

SIS Baseline Profile (SBP) Proposed Jurisdiction 2015



Table of Contents

SIS Baseline Profile (SBP) Proposed Jurisdiction 2015	1
1. Preface	4
1.1. Purpose	4
1.2. Principles.....	4
1.3. General Guidance	4
2. SIF Guidance.....	5
2.1. Objects Guidance.....	5
2.2. Messaging Guidance	7
2.2.1. Events (Pub/Sub)	8
2.2.2. Subscriber's guidance.....	9
2.2.3. Queries (Request/Response).....	9
2.2.4. Interpretation of queries.....	10
2.2.5. Service Paths	11
2.3. Immutable Primary/Alternate keys	12
2.3.1. Key for Tables:.....	12
3. Scenarios	12
Scenario Summary Matrix	13
3.0. Student & School Scenarios.....	18
3.0.1 School Scenarios	18

3.0.1.1 Linking School-centric SIS to existing jurisdiction systems	18
School Info	18
3.0.1.2 Add a New School	21
3.0.1.3 Close a School	23
3.0.1.4 Edit/Change School	24
3.0.1.5 Split Campus	25
3.0.1.6 Merge Schools	26
3.0.1.7 Merge Campus	27
3.0.1.8 Add Campus	28
3.0.1.9 Edit/Change Campus	29
3.0.1.9a Remove/Close Campus.....	30
3.0.1.9b Delete Campus	30
3.0.1.10 Transfer Campus.....	31
3.1 Student Scenarios	32
3.1.2 Future Enrollment	32
3.1.3 Future->Current Enrollment.....	36
3.1.4 New Current Enrollment	37
3.1.5 Year End Rollover	38
3.1.6 Exit School	40
3.1.7 Change Campus.....	41
3.1.8 Mid-year Promotion/Demotion	43
3.1.9 Change in Personal Details	45
3.1.10 Census Reporting:	47
3.2. Staff and Staff Assignment Scenarios	50
3.2.1 New Staff Member and School Assignment(s).....	50
3.2.2 Additional School Assignment (existing Staff Member).....	52
3.2.3 Change in Role (e.g. Promotion).....	54

3.2.4 Staff Member leaves School.....	56
3.2.5 Staff Member goes on leave (extended).....	57
3.2.6 Staff Member leaves Organisation.....	59
3.2.7 Change in Personal Details.....	61
3.3. Use Cases for the StudentContactPersonal and StudentContactRelationship.....	64
3.3.1. Student Contacts in Profiles and Learning Areas:	64
3.3.2. Student Attendance Follow-up:	64
3.3.3. SMS sending:	64
3.3.4 StudentContact and StudentContactRelationship Provisioning.....	64
3.3.4.1 New Student Contact and Relationship	65
3.3.4.2 New Relationship added to a Current Contact	68
3.3.4.3 Change to a StudentContact Person information	69
3.3.4.4 Change to StudentContact Relationship Info.....	71
3.3.4.5 Remove a StudentContact from a Student	73
3.4. Class scenarios	75
3.4.0. Scenario	75
3.4.0.1 Scenarios out of scope	75
3.4.1. Class-Centric information for publishing by an SIS	75
3.4.2. Provisioning Teaching Group periods.....	79
3.4.3. Teaching Group set up	80
3.4.4. Staff Joins Teaching Group	80
3.4.6. Staff Assignments Change in Teaching Group.....	84
3.4.7. Student Joins Teaching Group.....	86
3.4.8. Student Leaves Teaching Group.....	87
3.4.9. Period in Teaching Group Rescheduled	88
4. Optional Objects – may include	90
4.1. PersonPicture.....	90

1. Preface

This is the proposed update to the Student Information System Baseline Profile of SIF AU (SBP) for 2015. When completed and approved by the Data Standards Working Group this document will constitute the third iteration of this profile. The profile is updated to reflect the transition to SIF 3 Infrastructure.

1.1. Purpose

The SIS Baseline is based on an alliance between SIS vendors and Jurisdictions Members as a guide which would, once agreed, provide a set of SIF Objects and behaviours which will enable near 'plug and play' interoperability between a client and a server. The server will represent a Student Information System, at either the school or jurisdiction level, depending on whether the school is systemic or not.

It is also intended to provide advice to publishers of 'authoritative data sources' in relation to other key national initiatives such as the National Identity Management Framework (NIMF).

1.2. Principles

- Guidance, not regulation
- Drive simple interoperability, not just allow
- Consistency in expectation of behaviours between consumer and provider of information
- Adherence to SIF rules
- Working together for the benefit of all implementations but based on the experiences of real and specific implementations

1.3. General Guidance

The SBP has been built as a collaborative effort between key SIS stakeholders, schools software vendors who need information from the SIS and the SIF community including international vendors who have sought to find things that would be common to build a useful 'baseline'. The baseline is designed as both a school level (version 1.2) and a jurisdiction level (version 1.3) view of a data model and the associated behaviour which is listed in a range of scenarios. It is assumed that 'system of entry' or 'system of authority' of the data sources will be configured as the provider of these objects.

The SBP is designed to be built on as the need arises. For example, the addition of the PersonPicture object or the use of the Class schedule information in the TeachingGroup. These would not change any of the underlying baseline behaviours. In the Schools' version we have also considered the impact of scaling the scenarios up

to education jurisdiction level and the changes in behaviour that would be expected as an example of how the baseline could be grown into other contexts. The Jurisdiction version is an example of a scaled up context.

It is expected that the Baseline will develop and continue to grow over time, however it will remain the base to be built upon providing the reference point for others looking to start out in SIF.

Further to the successful development of an initial SIS Baseline Profile specifically for provisioning from de-centralised school systems, the jurisdiction view of provisioning into and out of a Jurisdiction has been developed. The SIS Baseline Profile was reviewed and further Use Cases, behaviours and functionality are described in this document to enable near 'plug and play' interoperability between a subscribing agent and a SIS or other jurisdiction system of record which has developed its agent up to the base line level for de-centralised and centralised systems.

2. SIF Guidance

2.1. Objects Guidance

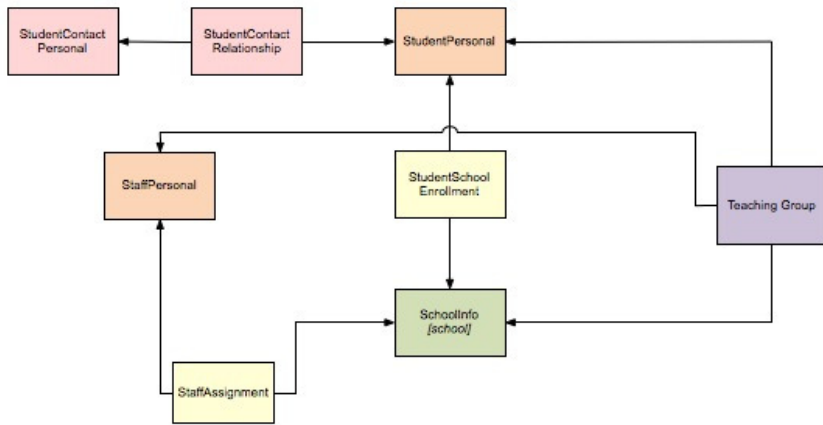
Objects in the Baseline

- SchoolInfo,
- StudentPersonal,
- StudentSchoolEnrollment
- StudentContactPersonal
- StudentContactRelationship
- StaffPersonal
- StaffAssignment
- TeachingGroup

(see Scenarios section for details on the Elements include for each object.)

Entity Relationship Diagram - School SBP v1.0

SIS Baseline Profile Objects

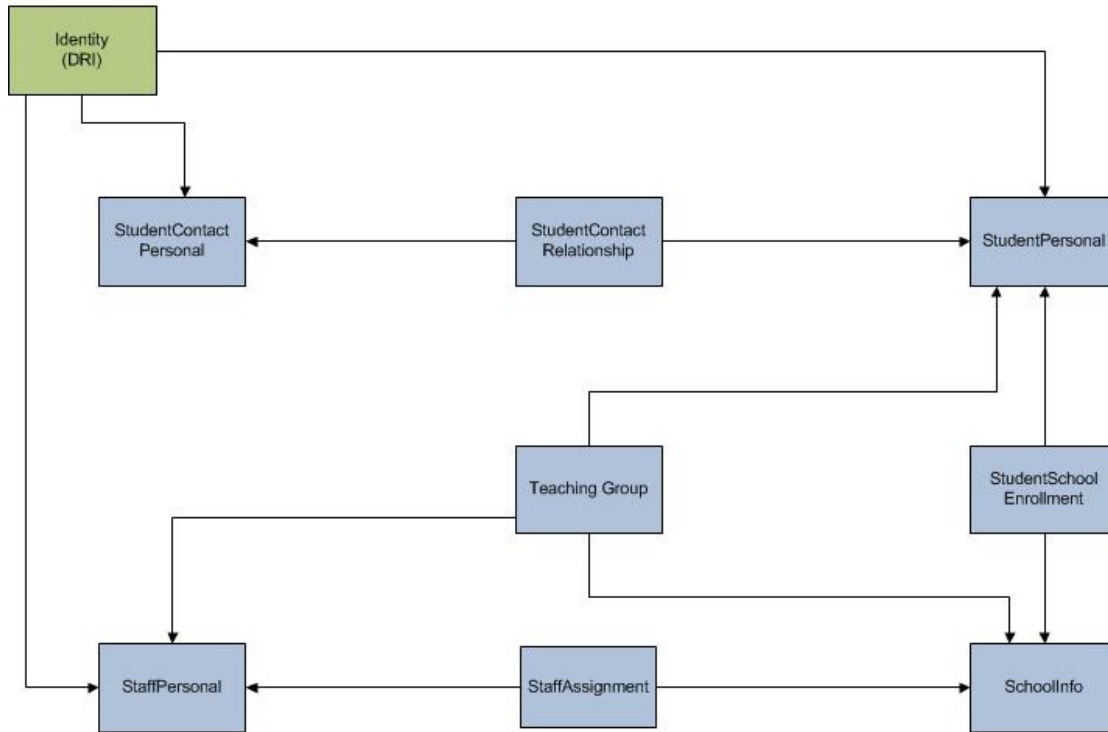


The SBP has simple Entity Relationship, the core identities are in the orange objects, the organisation is in green, the relationship between the school and individual entities are in yellow, the parents/guardians is in pink and the class, which links the teacher, student, school and subject together is in purple.

Entity Relationship Diagram - Jurisdiction SBP - Draft

The SBP Jurisdiction Draft Entity Diagram adds some entities to the current SBP to support Jurisdiction needs. The current school entities are in 'blue' additional jurisdiction entities are in 'green'.

SIS Jurisdiction Baseline Profile Objects v1.0



Missing Objects/Elements:

Where information required to support a particular business process is not included in the proposed Baseline, you should review other existing objects and elements in the SIF Implementation Specification (Australia) which contain the required information rather than attempting a custom extension to the baseline objects.

2.2. Messaging Guidance

2.2.1. Events (Pub/Sub)

The SIF 3 infrastructure allows for both event/subscribe and request/response modes of exchange. However event/subscribe relies on additional infrastructure being in place, and will likely only be implemented in systemic contexts.

For the identified Baseline elements, events are generated when they are updated. There are a number of ways to manage this, some systems keep a change log and have a process that checks the changes at intervals and then sends the messages when it detects a change. Another is to use database triggers at the time any change is made to the data.

To assist in identifying what fields are involved in an event;

- In an add event, all mapped fields associated with that event are assumed to be sent
- In a change event, only changed fields **HAVE** to be sent, however all fields **MAY** be sent
- In a delete event, no fields are sent.

Object	ADD Event	CHANGE Event	DELETE Event
SchoolInfo,	Create new record or update if already exists	Update record or create if not existing	Sent to rectify error. No scenario conditions
StudentPersonal,	Create new record or update if already exists	Update record or create if not existing.	Sent to rectify error. No scenario conditions
StudentSchoolEnrollment	Schoolinfo and Student personal sent beforehand. If not request, create new record or update if already exists	Update record or create if not existing	Sent to rectify error
StudentContactPersonal	Create new record or update if already exists	Update record or create if not existing	Sent to rectify error. No scenario conditions
StudentContactRelationship	StudentPersonal and StudentContactPersonal sent beforehand. If not, request. Create new record or update if already exists	Update record or create if not existing	Remove the relationship with Student, if the StudentContactPersonal record or the StudentPersonal record does not exist in the zone, then this represents a real 'Delete'.
StaffPersonal	Create new record or update if already exists	Update record or create if not existing	Sent to rectify error. No scenario conditions
StaffAssignment	Schoolinfo and Student personal sent beforehand. If not request. Create new record or update if already exists	Update record or create if not existing	Sent to rectify error. No scenario conditions
TeachingGroup	TBC	TBC	TBC

In the situation where a StudentContact Person is to be removed from a Student (3.3.4.5), a Delete event should be used in this situation as a Change event will not remove the contact person from the student.

2.2.2. Subscriber's guidance

The object provider should send all objects in order, and a subscriber should Request precedent objects in order; that is, producing object instances with Foreign key SIF Ref Ids: Primary key object instance should already exist on the zone. For the SBP these are School, Staff, Student. When first synchronising, (this is done by sending an unbounded request for an object), requests should be made for these precedent objects first.

The dependencies for the SBP are:

- StudentSchoolEnrollment requires both StudentPersonal and SchoolInfo objects
- StaffAssignment requires both StaffPersonal and SchoolInfo objects
- Teaching Group requires SchoolInfo, StaffPersonal and StudentPersonal
- StudentContactRelationship requires StudentPersonal and StudentContactPersonal

However in the unlikely case of objects being 'out of order' the preferred behaviour is;

- The Subscribers should request the missing object.
- If there is an 'Add' for an existing object then the subscriber should automatically assume it's a Change
- If there is a Change for a non-existing object subscriber should assume it's an Add.
- If the subscriber does not perform these tasks it must at least alert the operator to the error condition for manual intervention.

2.2.3. Queries (Request/Response)

All SIS shall support unbounded Queries for the baseline objects. This is known as Request/Response model within SIF where a query is sent to the SIS agent from a client and the SIS sends a response. Only a limited subset of queries (bounded requests) will be supported as part of the baseline. The table below outlines these.

Object	Unbounded	Key/Mandatory	Other Queries
SchoolInfo	All Campus returned.	Return single object based on RefId	
StudentPersonal	All Current* Students	Return single object based on RefId Return single object based on LocalId	

StudentSchoolEnrollment	All Current* Student Enrolments	Return single object based on RefId	
StudentContactPersonal	All Current* Students contacts	Return single object based on RefId	
StudentContactRelationship	All Current* Students contacts	Return single object based on RefId	
StaffPersonal	All Employed Staff	Return single object based on RefId	
StaffAssignment	All Assignments without an end date.	Return single object based on RefId	
TeachingGroup	All Teaching groups.	Return single object based on RefId	<p>All Teaching groups for which TeachingGroup/TeacherList/TeachingGroupTeacher/StaffLocalId matches X</p> <p>All Teaching groups for which TeachingGroup/StudentList/TeachingGroupStudent/StudentLocalId matches X (Strongly recommended)</p> <p>All Teaching groups for which TeachingGroup/TeacherList/TeachingGroupTeacher/StaffPersonalRefId matches X and TeachingGroup/SchoolInfoRefId matches Y (Strongly recommended)</p> <p>All Teaching groups for which TeachingGroup/TeacherList/TeachingGroupStudent/StudentPersonalRefId matches X and TeachingGroup/SchoolInfoRefId matches Y</p>

*See Currency below

2.2.4. Interpretation of queries

"Currency" query definition

For unbounded queries where timeframe exists for 'currency' in the object (e.g. StudentSchoolEnrollment) then an 'unbounded' request shall mean 'all current and confirmed next year's students'. Eg. Current enrolment year plus one. This is used as part of the Census gathering requirements of schools.

For Subscribing systems which only want current enrolled year students; e.g. For populating LMS -- there is a need to know who is a currently enrolled student, rather than a future. To achieve this the subscriber MUST ignore all enrollments with the "TimeFrame" set to other than "Current".

This is because many other schools systems include the next year's students for the planning and census information when they are determining current students and given that the Timeframe is a coded element it is simple for the subscribing agent to drop the students other than those set to current, and the same zone can then be used for many more purposes.

Definition of students who should be included in a request for a 'current student' query

1. All confirmed(*) students for the current enrolled year.
 2. All confirmed(*) students for the subsequent enrollment year. (year +1)
 3. If there is a Change for a non-existing object subscriber should assume it's an Add.
- (*) confirmed - the definition of which would be determined by the SIS. e.g. paid enrolment fee, accepted a place, filled out enrolment details etc.

Definition of students who should be included in a request for a 'future student' query

1. All students who have been confirmed (*) as enrolling for the subsequent school year. (year +1)
- (*) confirmed - the definition of which would be determined by the SIS. e.g. paid enrolment fee, accepted a place, filled out enrolment details etc.

2.2.5. Service Paths

Service Paths are part of the SIF specification, but the particular service paths available from a server are an implementation decision. Under this profile, the following service paths are expected to be supported:

Object	Unbounded	Key/Mandatory	School
SchoolInfo	/SchoolInfos	/SchoolInfos/{RefId}	
StudentPersonal	/StudentPersonals	/StudentPersonals/{RefId}	/SchoolInfos/{RefId}/StudentPersonals
StudentSchoolEnrollment	/StudentSchoolEnrollments	/StudentSchoolEnrollments/{RefId}	/SchoolInfos/{RefId}/StudentSchoolEnrollments
StudentContactPersonal	/StudentContactPersonals	/StudentContactPersonals/{RefId}	/SchoolInfos/{RefId}/StudentPersonals
StudentContactRelationship	/StudentContactRelationships	/StudentContactRelationships/{RefId}	/SchoolInfos/{RefId}/StudentContactPersonal
StaffPersonal	/StaffPersonals	/StaffPersonals/{RefId}	/SchoolInfos/{RefId}/StaffPersonals
StaffAssignment	/StaffAssignments	/StaffAssignments/{RefId}	/SchoolInfos/{RefId}/StaffAssignments
TeachingGroup	/TeachingGroups	/TeachingGroups/{RefId}	/SchoolInfos/{RefId}/TeachingGroups

Object	Other	
StudentPersonal	/StudentPersonals/{RefId}/StudentContactPersonals	Join on StudentPersonal@RefId = StudentContactPersonal/StudentPersonalRefId
TeachingGroup	/TeachingGroup/{RefId}/StudentPersonals	Join on StudentPersonal@RefId = TeachingGroup/StudentList/TeachingGroupStudent/StudentPersonalRefId
TeachingGroup	/TeachingGroup/{RefId}/StaffPersonals	Join on StaffPersonals@RefId = TeachingGroup/TeacherList/TeachingGroupTeacher/StaffPersonalRefId

School queries are only required in school systems, where information about multiple schools is available from the SIS.

The scope of records returned is expected to follow the currency definitions given above.

2.3. Immutable Primary/Alternate keys

Under SIF AU 1.4, all objects have RefIds which are treated as primary keys, and which must not be changed through the object's lifetime. Primary/alternate keys are no longer indicated in this version of the profile.

2.3.1. Key for Tables:

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
Identifies attribute of element or object key	SIF (Australia) r1.x Implementation Specification Object/Element Name	As identified by the SIF Implementation (Australia) 1.x	Identifies Baseline requirements for SIF Implementation (Australia) 1.x optional elements M = Mandatory S = Strongly Suggested C = Conditional R = Repeatable O = Optional N = Not Used	Object/Element SIF Implementation (Australia) 1.x description	Identifies Baseline SIS business rules for object/element

3. Scenarios

The scenarios identified here are considered 'typical'. They do not necessarily map exactly to the specific needs of any given jurisdiction. Therefore it is expected that some specific implementations may wish to extend this document to exactly match the needs.

All said; the intent of this document is as a baseline such that there is minimal difference between implementations. This reduces the rework required by suppliers and instils consumer confidence in the ability of SIF to work out-of-the-box.

Scenario Summary Matrix

The following is a list of the scenarios that comprise an Australian SBP. This profile is currently under development and when formalized, will be available at: specification.sifassociation.org/implementation/au/1.3/files/SBP1.3.pdf

Scenario	AU Object element/behavior	Message	Behavior	SBP Rules
3.0.1.1 Linking School-centric SIS to existing jurisdiction systems	SchoolInfo	Respond	Respond	A SIS must be able to respond to an unbounded SchoolInfo request. It should include information about currently open schools and campuses only.
3.0.1.2 Add a New School	SchoolInfo	Event	Create	Entity_Open date must be published with open date
3.0.1.3 Close a School	SchoolInfo	Event	Change	Entity_Close date must be published with date of closure
3.0.1.4 Edit/Change School	SchoolInfo	Event	Change	Changed Elements published with changes
3.0.1.5 Split Campus	SchoolInfo	Event	Change	SchoolInfo published with any changed elements
	SchoolInfo	Event	Create	Entity_Open date must be published with information about the new Campus
3.0.1.6 Merge Schools	SchoolInfo	Event	Change	Populate Entity_Close Date on the campus(es) being merged
	SchoolInfo	Event	Change or Create	Elements updated if a change, or, if a new School is created, Entity_Open date must be published
3.0.1.7 Merge Campus	SchoolInfo	Event	Change	Entity_Close date to be published with close date of the closing campus
	SchoolInfo	Event	Change	SchoolInfo elements to be updated with changed campus details
3.0.1.8 Add Campus	SchoolInfo	Event	Create	Entity_Open date must be published with open date
3.0.1.9 Edit/Change Campus	SchoolInfo	Event	Change	Changes made should be published
3.0.1.9a Remove/Close Campus	SchoolInfo	Event	Change	Entity_Close date must be published with date of closure
3.0.1.9b Delete Campus	SchoolInfo	Event	Delete	Entity_Close date should be published even if record was generated in error

3.0.1.10 Transfer Campus	SchoolInfo	Event	Change	Entity_Close date published with close date of the closing campus
	SchoolInfo	Event	Create	Entity_Open date must be published for transferred campus
3.1.2. Future Enrollment	StudentPersonal	Event	Create	Event to be published by SIS. It is recommended, (but not required), that the event should be published in near real-time.
	StudentSchoolEnrollment	Event	Create	SchoolYear: As this is a 'Future' enrolment, this value may be greater than the current SchoolYear This event is to be published after the StudentPersonal event TimeFrame: ='F' for future
3.1.3. Future->Current Enrollment	StudentSchoolEnrollment	Event	Change	Both StudentPersonal and SchoolInfo must already have been published to the zone by the designated 'provider' Change value of TimeFrame to 'C' for Current The TimeFrame value changes when a student becomes current.
3.1.4. New Current Enrollment	StudentPersonal	Event	Create	StudentPersonal needs to be created.
	StudentSchoolEnrollment	Event	Create	SchoolYear: As this is a "Current" enrolment, this value must be equal to the current SchoolYear TimeFrame: ='C' for Current EntryDate: As this enrolment is current the EntryDate must be the same or less than the day of entry of the information into the SIS
3.1.5. Year End Rollover	StudentSchoolEnrollment	Event	Change	TimeFrame: = 'H' for Historical RecordClosureReason: = EndOfYear
	StudentSchoolEnrollment	Event	Create	TimeFrame: ='C' for Current SchoolYear: As this is a "Current" enrolment, this value must be equal to the current SchoolYear. It is recommended this event is published after the 'Change' SSE event. This can either be in batches or pairs.
3.1.6. Exit School	StudentSchoolEnrollment	Event	Change	TimeFrame: = 'H' for Historical
3.1.7. Change Campus	StudentSchoolEnrollment	Event	Change (for existing object)	RecordClosureReason: = CampusExit ExitType: Transfer campus of the same school TimeFrame='H' for Historical
	StudentSchoolEnrollment	Event	Create (for new object)	TimeFrame: ='C' for Current SchoolYear: As this is a "Current" enrolment, this value must be equal to the current SchoolYear.

				EntryType: Transfer from a different campus of the same school
3.1.8. Mid-year Promotion/Demotion	StudentSchoolEnrollment	Event	Change	YearLevel: - Change with new level PromotionInfo/PromotionStatus: Should be updated with value from value list.
	StudentPersonal	Event	Change	YearLevel: - Change with new level if previously published.
3.1.9 Change in Personal Details	StudentPersonal	Event	Change	Update the changed elements.
3.1.10 Census Reporting	StudentPersonal	Event	Change	Demographic information intended for Census reporting has been added to Demographic information into the standard jurisdiction SBP model for Parents and Students. This information should be updated and published as a Change Event when Parent Information is not published.
3.2.1. New Staff Member and School Assignment(s)	StaffPersonal	Event	Create	EmploymentStatus: Set to 'Active'
	StaffAssignment	Event	Create	PrimaryAssignment: Set to 'Y' for the primary staff assignment
3.2.2 Additional School Assignment (existing Staff Member)	StaffAssignment (pre-existing)	Event	Change	PrimaryAssignment: Modify where required to reflect status of the new assignment.
	StaffAssignment (new)	Event	Create	PrimaryAssignment: Set to 'Y' for the primary staff assignment
3.2.3 Change in Role (e.g. Promotion)	StaffAssignment (pre-existing)	Event	Change	StaffActivity/Code – Update PrimaryAssignment: JobEndDate:
	StaffAssignment (new)	Event	Create	PrimaryAssignment: Set to 'Y' for the primary staff assignment
3.2.4 Staff Member leaves School	StaffAssignment	Event	Change – to end a role	JobEndDate: is populated for each active StaffAssignment in the school but cannot be pre-dated. Where the school is the organization go to scenario: 3.2.6
3.2.5 Staff Member goes on leave (extended)	StaffPersonal	Event	Change	EmploymentStatus: Set to "on leave". No current StaffAssignments should exist but if they do they should be ignored in the zone.
3.2.6 Staff Member leaves Organisation	StaffPersonal	Event	Change	EmploymentStatus: set to "Inactive"
	StaffAssignment	Event	Change	For each StaffAssignment JobEndDate: is populated with Ending Date which cannot be pre-dated.
3.2.7 Change in Personal Details	StaffPersonal	Event	Change	Update the changed elements.
3.3.4.1 New Student Contact and Relationship	StudentContactPersonal	Event	Create	At least one student whose contact this is must have been previously published.
	StudentContactRelationship	Event	Create	A new object is published for each student that the Student Contact has a relationship with.
3.3.4.2 New Relationship added to a Current Contact	StudentContactRelationship	Event	Create	Both the Student and the StudentContact must have been previously published. This new object establishes a new relationship between the two entities.
3.3.4.3 Change to a	StudentContactPersonal	Event	Change	Update changed elements.

StudentContact Person information				
3.3.4.4 Change to StudentContact Relationship Info	StudentContactRelationship	Event	Change	Update changed details including updating ContactFlags where contact priorities have changed.
3.3.4.5 Remove a StudentContact from a Student	StudentContactRelationship	Event	Delete	This removes the relationship between a Student and a Contact and represents a “true delete” to indicate the severing of the relationship.
3.4.1. Class-Centric information for publishing by an SIS	TeachingGroup	Respond	Respond	A SIS must be able to respond to a request for all TeachingGroup objects within the zone relating to a particular student or staff member. The StudentPersonal and StaffPersonal objects referenced by the TeachingGroup must already be published (per provisioning scenarios below). If the SIS is reporting on the output of a Timetabling system, the TimeTableCell objects referenced by TeachingGroup must already be published.
3.4.3. Teaching Group set up	TeachingGroup	Event	Create	Either TeachingGroup/StudentList or TeachingGroup/TeacherList contain at least one valid StudentLocalId or StaffLocalId instance TeachingGroup/TeachingGroupPeriodList contains at least one non-empty instance of TeachingGroupPeriod
3.4.4. Staff Joins Teaching Group	TeachingGroup	Event	Change	TeacherList: Add StaffLocalId TeachingGroupPeriodList/TeachingGroupPeriod: Remove StaffLocalId to each relevant TeachingGroupPeriod Re-publish lists. Any changes in TeacherList or TeachingGroup PeriodList mean that the Object needs to be published with the complete updated list.
3.4.5. Staff Leaves Teaching Group	TeachingGroup	Event	Change	TeacherList: Remove StaffLocalId TeachingGroupPeriodList/TeachingGroupPeriod: Remove StaffLocalId to each relevant TeachingGroupPeriod Re-publish lists. Any changes in TeacherList or TeachingGroupPeriodList mean that the Object needs to be published with the complete updated list.
3.4.6. Staff Assignments Change In Teaching Group	TeachingGroup	Event	Change	TeacherList: TeachingGroupPeriodList: Changes in the Staff member or their Association with a TeachingGroup may change. Re-publish lists. Any changes in TeacherList or TeachingGroupPeriodList mean that the Object needs to be published with the complete updated list.

3.4.7. Student Joins Teaching Group	TeachingGroup	Event	Change	<p>StudentList: Add Student LocalId to list;</p> <p>Re-publish lists. Any changes in StudentList meant that the Object needs to be published with the complete updated list.</p>
3.4.8. Student Leaves Teaching Group	TeachingGroup	Event	Change	<p>StudentList: Remove Student LocalId from list;</p> <p>Re-publish lists. Any changes in StudentList meant that the Object needs to be published with the complete updated list.</p>
3.4.9. Period in Teaching Group Rescheduled	TeachingGroup	Event	Change	<p>TeachingGroupPeriodList: Add/Remove TeachingGroupPeriod from TeachingGroupPeriodList.</p> <p>Re-publish lists. Any changes in TeachingGroupPeriodList mean that the Object needs to be published with the complete updated list.</p>

DRAFT until approved

3.0. Student & School Scenarios

3.0.1 School Scenarios

3.0.1.1 Linking School-centric SIS to existing jurisdiction systems

This scenario is a pre-requisite for many other scenarios. It deals with the need to ensure that SchoolInfoRefId is shared zone-wide for scenarios that involve school-centric SIS to interoperate with existing systems held at jurisdiction level. It is assumed that typically the SIS is the 'System of Entry' for the SIF RefId (which would probably be generated automatically rather than entered manually, so 'System of Creation' might be more appropriate). It is also assumed that basic school information (name, address etc) is already known centrally. Furthermore, it is assumed that out-of-band processes will ensure school information in central jurisdiction systems is aligned with that held in school centric SIS.

(That's not to say that the out-of-band processes should not be brought into these guidelines in the future – they're not just not part of the scope at this stage).

Given the assumptions above the following prerequisites and in-process requirements are specified

Pre-requisites:

Non-SIF: Basic school data, specifically 'LocalId' is already shared between known systems
SIF: n/a

In-process requirements:

Object	Message	Behaviour	Choreography
SchoolInfo	Respond	Respond	A SIS must be able to respond to an unbounded SchoolInfo request. It should include information about currently open schools and campuses only.

School Info

Data Elements that need to be sent as a minimum – other 'O' optional elements in the full SIF Specification can be published, but as a minimum subscriber's typically need to consume the following:

TABLE S1.0

Element/Attribute	Char	Baseline Char	Description	Business Rules
RefId	M	M	The GUID that identifies this school/campus	
LocalId	M	M	The locally-assigned identifier for this school/campus	
StateProvinceId	O	S	The state-assigned identifier for this school/campus	
CommonwealthId	O	S	Commonwealth identifier for this school/campus.	
ACARAId	O	S	ACARA identifier for this school/campus.	
OtherIdList	O	O	Lists all "other" identifiers associated with the school/campus	This element only available from version 1.3
OtherIdList/OtherId	MR	MR	Lists an "other" identifier associated with the school.	OtherId is Mandatory if the OtherIdList is being published. Only available from version 1.3
SchoolName	M	M	Name of school/campus.	
Campus (and associated elements)	O	C	Campus identifier	Must be included if the school is defined as multi-campus. (Used to identify type of Campus)
Campus/AdminStatus	M	M	Is this Campus the Administration Campus?	Must be included if the school is defined as multi-campus
Campus/CampusType	O	O	Type of campus.	Supply if different from the School Type.
Campus/ParentSchoolId	C	CS	Parent School Identifier	Parent School Id needs to be supplied if this Campus is not the Admin Campus. This number should be the state-assigned identifier for this school or the locally-assigned identifier for this school. Only available from 1.3 onwards.
OperationalStatus	O	S	Operational condition of a school.	
SchoolType	O	S	An indication of the level of the educational institution	
SchoolSector	M	M	Government or Non Government	
SchoolFocusList	O	O	List containing the education focus codes of the campus/school.	

SchoolFocusList/SchoolFocus	MR	MR	Focus of School eg Regular, Special, Vocational	Mandatory if SchoolFocusList is being provided.
IndependentSchool	O	S	Indicator that this school is an independent school	
SchoolGroupList	O	O		
SchoolGroupList/SchoolGroup	MR	MR	Repeatable element containing a local identifier to identify a loosely connected group of schools. eg. YarraValley Cluster	Mandatory if SchoolGroupList is being provided.
ARIA	O	O	Accessibility/Remoteness Indicator of Australia.	
AddressList	O	S		AddressList is an Optional List of Addresses associated with a School. It is strongly suggested that at least one address is sent by the authoritative Jurisdiction.
AddressList/Address	MR	MR		If an Address is supplied it is mandatory to send the Type and Role Attributes. e.g. <Address Type="0123" Role="012B">
AddressList/Address/StateProvince	C	S	The state or province code eg NSW, Vic, ACT	Only supply if in Australia
AddressList/Address/City	M	M	The city part of the address	
AddressList/Address/PostalCode	M	M	The ZIP/Postal code	
AddressList/Address/Street/StreetNumber	O	S	The address number assigned to the building	
AddressList/Address/Street/StreetName	O	S	The name of the street	
AddressList/Address/GridLocation/Longitude	O	O	Longitude of school location	
AddressList/Address/GridLocation/Latitude	O	O	Latitude of school location	
AddressList/Address/SchoolGeographicLocation	O	O	School Location from MCEETYA.	
Entity_Open	O	S	Open Date of Entity	When a school/campus is opened its Open Date is required to be sent when it can be supplied.
Entity_Close	O	S	Close Date of Entity	When a school/campus is Closed its date of closure is required to be sent when it can be supplied.

A SIS may contain information for a multi-campus school. In this case, a distinct SchoolInfo object (and therefore a distinct SchoolInfoRefId) must be returned for each Campus.

A SIS may also contain information for multiple schools. Implementation specific rules will govern whether all known schools should be included in the overall Response. These rules may be determined, for example, by the zone design.

3.0.1.2 Add a New School

Event	Description - Trigger	User Input	Agent Action - Publisher
Add new school	New school added at Jurisdiction.	Jurisdiction staff adds a new school to its system	Agent detects the change and sends an ADD event for SchoolInfo (including Entity_Open date)

Object Behaviour

Object	Message	Behaviour	Choreography
SchoolInfo	Event	Add	An Entity Open date must be published. If actual Open date not available, date added to the authoritative source.

SchoolInfo

Suggested Data Elements to be supplied with an "Add School"

Element/Attribute	Char	Baseline Char	Description	Business Rules
RefId	M	M	The GUID that identifies this school/campus	
LocalId	M	M	The locally-assigned identifier for this school/campus	
StateProvinceId	O	S	The state-assigned identifier for this school/campus	
CommonwealthId	O	S	Commonwealth identifier for this school/campus.	
ACARAId	O	S	ACARA identifier for this school/campus.	
OtherIdList	O	O	Lists all "other" identifiers associated with the school/campus	This element only available from version 1.3
OtherIdList/OtherId	MR	MR	Lists an "other" identifier associated with the school.	OtherId is Mandatory if the OtherIdList is being published. Only available from version 1.3
SchoolName	M	M	Name of school/campus.	

Campus (and associated elements)	O	C	Campus identifier	Must be included if the school is defined as multi-campus. (Used to identify type of Campus)
Campus/AdminStatus	M	M	Is this Campus the Administration Campus?	Must be included if the school is defined as multi-campus
Campus/CampusType	O	O	Type of campus.	Supply if different from the School Type.
Campus/ParentSchoolId	C	CS	Parent School Identifier	Parent School Id needs to be supplied if this Campus is not the Admin Campus. This number should be the state-assigned identifier for this school or the locally-assigned identifier for this school. Only available from 1.3 onwards.
OperationalStatus	O	S	Operational condition of a school.	
SchoolType	O	S	An indication of the level of the educational institution	
SchoolSector	M	M	Government or Non Government	
SchoolFocusList/SchoolFocus	MR	MR	Focus of School eg Regular, Special, Vocational	Mandatory if SchoolFocusList is being provided.
SchoolFocus	MR	MR	Focus of School eg Regular, Special, Vocational	
IndependentSchool	O	S	Indicator that this school is an independent school	
SchoolGroupList	O	O		
SchoolGroupList/SchoolGroup	MR	MR	Repeatable element containing a local identifier to identify a loosely connected group of schools. eg. YarraValley Cluster	Mandatory if SchoolGroupList is being provided.
ARIA	O	O	Accessibility/Remoteness Indicator of Australia.	
AddressList	O	S		AddressList is an Optional List of Addresses associated with a School. It is strongly suggested that at least one address is sent by the authoritative Jurisdiction.
AddressList/Address	MR	MR		If an Address is supplied it is mandatory to send the Type and Role Attributes. e.g.

					<Address Type="0123" Role="012B">
	AddressList/Address/StateProvince	C	S	The state or province code eg NSW, Vic, ACT	Only supply if in Australia
	AddressList/Address/City	M	M	The city part of the address	
	AddressList/Address/PostalCode	M	M	The ZIP/Postal code	
	AddressList/Address/Street/StreetNumber	O	S	The address number assigned to the building	
	AddressList/Address/Street/StreetName	O	S	The name of the street	
	AddressList/Address/GridLocation/Longitude	O	O	Longitude of school location	
	AddressList/Address/GridLocation/Latitude	O	O	Latitude of school location	
	AddressList/Address/SchoolGeographicLocation	O	O	School Location from MCEETYA.	
	Entity_Open	O	S	Open Date of Entity	When a school/campus is opened its Open Date is required to be sent when it can be supplied. If actual Open date not available, date added to the authoritative source must be supplied.

3.0.1.3 Close a School

Event	Description - Trigger	Agent Action - Publisher
Close existing school	A Jurisdiction closes a school. When a school is closed - the Entity_Close date needs to be updated.	Agent detects the change and sends Entity_Close date with the date of closure.

Object Behaviour

Object	Message	Behaviour	Choreography
SchoolInfo	Event	Change	An Entity_Close date must be published. If date of closure not available, date deprecated in the authoritative source should be provided.

SchoolInfo

Element/Attribute	Char	Baseline Char	Description	Business Rules
RefId	M	M	The GUID that identifies this school/campus	
LocalId	M	M	The locally-assigned identifier for this	

				school/campus	
	Entity_Close	O	S	Close Date of Entity	When a school/campus is Closed its date of closure is required to be sent when it can be supplied. If date of closure not available, date deprecated in the authoritative source should be provided.

3.0.1.4 Edit/Change School

Event	Trigger	Agent Action - Publisher
Edit/change existing school	A change is made to any field in the source system which is mapped to the SchoolInfo object a change event is sent with only that information in the event.	Agent detects the change and sends a Change event for SchoolInfo

Object Behaviour

Object	Message	Behaviour	Choreography
SchoolInfo	Event	Change	Changed Elements published with changes

SchoolInfo

Refer to TABLE S1.0

Elements in this table to be supplied with a "Edit/Change" School when the elements change together with the following identifiers required for consistency.

Element/Attribute	Char	Baseline Char	Description	Business Rules
RefId	M	M	The GUID that identifies this school/campus	
LocalId	M	M	The locally-assigned identifier for this school/campus	
StateProvinceId	O	S	The state-assigned identifier for this school/campus	
CommonwealthId	O	S	Commonwealth identifier for this school/campus.	
ACARAId	O	S	ACARA identifier for this school/campus.	

3.0.1.5 Split Campus

Event	Trigger	Agent Action - Publisher
Split an existing campus	An existing campus of a school splits into two - a Change event needs to be issued for the original campus - if any of the elements have changed or, if it was a single campus school and is now a dual campus, the Campus container needs to be updated. AND an Add event needs to be issued for the new campus with the additional information..	The original campus SchoolInfo record will be treated as a 'Change of School details'. The new campus with the SchoolInfo record sent as an 'Add' will be treated as a New campus. The School may change from single campus to multicampus, requiring Campus/AdminStatus to be provided.

Object Behaviour

Object	Message	Behaviour	Choreography
SchoolInfo	Event	Change	SchoolInfo published with any changed elements (as per 3.0.1.4)
SchoolInfo	Event	Add	An Entity Open date of the newly created Campus must be published. If actual Open date not available, date added to the authoritative source. (as per 3.0.1.2)

SchoolInfo

Refer to TABLE S1.0

Change: Elements in this table to be supplied with a "Edit/Change" when the elements change together with the following identifiers required for consistency.

Add: Mandatory elements in TABLE S1.0 to be supplied with an "Add" School

	Element/Attribute	Char	Baseline Char	Description	Business Rules
	RefId	M	M	The GUID that identifies this school/campus	
	LocalId	M	M	The locally-assigned identifier for this school/campus	
	StateProvinceId	O	S	The state-assigned identifier for this school/campus	
	CommonwealthId	O	S	Commonwealth identifier for this school/campus.	
	ACARAId	O	S	ACARA identifier for this school/campus.	
	SchoolName	M	M	Name of school/campus.	

Campus (and associated elements)	O	C	Campus identifier	Must be published in a split of a campus if any change is detected in sub-elements.
Campus/AdminStatus	M	M	Is this Campus the Administration Campus?	Must be published in a split of a campus if any change are detected.
Campus/CampusType	O	O	Type of campus.	Must be published in a split of a campus if any change are detected.
Campus/ParentSchoolId	C	CS	Parent School Identifier	Parent School Id needs to be supplied if this Campus is not the Admin Campus. This number should be the state-assigned identifier for this school or the locally-assigned identifier for this school. Must be published in a split of a campus if any change are detected Only available from 1.3 onwards.

3.0.1.6 Merge Schools

Event	Trigger	Agent Action - Publisher
Merge existing schools	A Jurisdiction merges two existing schools.	<p>a. The two schools are merged into one currently existing school. The school which is closed publishes a <u>Delete/Close of a School</u>. As per 3.0.1.3 The school which remains publishes a <u>Change of School details</u> as per 3.0.1.4</p> <p>b. Both the two existing schools are closed and a new school is created which may or may not be multicampus. Both existing schools trigger <u>Delete/Close of a School</u> events. As per 3.0.1.3 The newly created school triggers an <u>Add School</u> event. As per 3.0.1.2</p>

Object Behaviour

Object	Message	Behaviour	Choreography
SchoolInfo	Event	Change	Populate Entity_Close Date on the school(s) being merged. Campus/AdminStatus must be updated if changed.
SchoolInfo	Event	Change or Add	Elements updated if a change to an existing school or if a new School is created - Add Entity_Open date must be published with date of opening or date of entry into authoritative source.

3.0.1.7 Merge Campus

Refer to 3.0.1.6 Merge School

Event	Trigger	Agent Action - Publisher
Merge existing campuses	Jurisdiction merges existing campuses either by merging one into the other or closing both and creating a new entity.	The campus which is closed should perform a <u>Delete/Close of a School</u> . The campus which remains needs to be updated with <u>Change of School details</u> The School may change from multi campus to single campus, this may require <u>Campus/AdminStatus</u> to be modified

Object	Message	Behaviour	Choreography
SchoolInfo	Event	Change	Populate Entity_Close Date on the campus(es) being merged
SchoolInfo	Event	Change	SchoolInfo elements to be updated with changed campus details

SchoolInfo

Refer to TABLE S1.0

Change: Elements in this table to be supplied with a "Edit/Change" when the elements change together with the following identifiers required for consistency.

Add: Mandatory elements in TABLE S1.0 to be supplied with an "Add" School/Campus

Element/Attribute	Char	Baseline Char	Description	Business Rules
RefId	M	M	The GUID that identifies this school/campus	
LocalId	M	M	The locally-assigned identifier for this school/campus	
StateProvinceId	O	S	The state-assigned identifier for this school/campus	
CommonwealthId	O	S	Commonwealth identifier for this school/campus.	
ACARAId	O	S	ACARA identifier for this school/campus.	
SchoolName	M	M	Name of school/campus.	
Campus (and associated elements)	O	C	Campus identifier	Must be published in a split of a campus if any change is detected in sub-elements.
Campus/AdminStatus	M	M	Is this Campus the Administration Campus?	Must be published in a split of a campus

					if any change are detected.
	Campus/CampusType	O	O	Type of campus.	Must be published in a split of a campus if any change are detected.
	Campus/ParentSchoolId	C	CS	Parent School Identifier	Parent School Id needs to be supplied if this Campus is not the Admin Campus. This number should be the state-assigned identifier for this school or the locally-assigned identifier for this school. Must be published in a split of a campus if any change are detected Only available from 1.3 onwards.

3.0.1.8 Add Campus

Refer to 3.0.1.2 Add a New School

Event	Trigger	Agent Action - Publisher
Add campus	Jurisdiction staff add a new campus to an existing school. Where a new Jurisdiction Id is generated for a new campus an Add SchoolInfo Event is to be generated.	Agent detects the change and sends an ADD event for SchoolInfo (including Entity_Open date) The CommonwealthId should be updated with the DEEWR Id where available. If this "Add" of a new campus changes the status of other campus(es) of the school then a "Change" events of the relevant campus elements needs to bepublished.

Object Behaviour

Object	Message	Behaviour	Choreography
SchoolInfo	Event	Add	An Entity Open date must be published. If actual Open date not available, date added to the authoritative source.
SchoolInfo	Event	Change	If the addition of a new Campus changes the status of Campus information held against other Campus(es) of the school a Change Event needs to update any elements within the Campus container. E.g. AdminStatus, ParentSchoolId etc.

SchoolInfo

Refer to TABLE S1.0

Add: Mandatory elements in TABLE S1.0 to be supplied with an "Add" School together with Campus information.

Element/Attribute	Char	Baseline Char	Description	Business Rules
RefId	M	M	The GUID that identifies this school/campus	
LocalId	M	M	The locally-assigned identifier for this school/campus	
StateProvinceId	O	S	The state-assigned identifier for this school/campus	
CommonwealthId	O	S	Commonwealth identifier for this school/campus.	
ACARAId	O	S	ACARA identifier for this school/campus.	
SchoolName	M	M	Name of school/campus.	
Campus (and associated elements)	O	C	Campus identifier	Must be published in a add of a campus or if any change is detected in sub-elements of an existing campus.
Campus/AdminStatus	M	M	Is this Campus the Administration Campus?	Must be published with an add campus and/or if any changes are detected.
Campus/CampusType	O	O	Type of campus.	Must be published in a split of a campus and/or if any changes are detected.
Campus/ParentSchoolId	C	CS	Parent School Identifier	Parent School Id needs to be supplied if this Campus is not the Admin Campus. This number should be the state-assigned identifier for this school or the locally-assigned identifier for this school. Must be published in an add campus and/or if any change are detected Only available from 1.3 onwards.

3.0.1.9 Edit/Change Campus

Refer to 3.0.1.4 Edit/Change School

Event	Trigger	Agent Action - Publisher
-------	---------	--------------------------

Change campus	An existing campus changes its details. ie change of address Jurisdiction staff change details of a campus.	Agent detects the change and sends a CHANGE event for SchoolInfo, if change of one campus triggers a change in another Campus(es) AdminStatus or ParentSchoolId, the second Campus a CHANGE event needs to be triggered for the effected Campus.
----------------------	--	--

Object Behaviour

Object	Message	Behaviour	Choreography
SchoolInfo	Event	Change	Changes made should be published

3.0.1.9a Remove/Close Campus

Refer to 3.0.1.3 Close a School

Event	Trigger	Agent Action - Publisher
Remove campus	Jurisdiction staff close a campus.	Agent populates the Entity_Close date and sends a CHANGE event for SchoolInfo AdminCampus Status needs to be updated if the Admin Campus changes.

Object Behaviour

Object	Message	Behaviour	Choreography
SchoolInfo	Event	Change	Entity_Close date must be published

3.0.1.9b Delete Campus

Event	Trigger	User Input	Agent Action - Publisher
Delete campus	A campus is created in error - physical deletion of jurisdiction record.	Jurisdiction staff physically delete an incorrect campus record.	Agent detects the change and sends a DELETE event for SchoolInfo

Object Behaviour

Object	Message	Behaviour	Choreography
SchoolInfo	Event	Delete	Entity_Close date should be published even if record was generated in error

SchoolInfo

Refer to TABLE S1.0: Only RefId required to be sent with a "Delete" event.

Element/Attribute	Char	Baseline Char	Description	Business Rules
RefId	M	M	The GUID that identifies this school/campus	

3.0.1.10 Transfer Campus

Event	Trigger	User Input	Agent Action - Publisher
Transfer campus	A campus of a school is moved to another school.	Campus information is transferred to another Campus	a) Agent populates the Entity_Close date and sends a CHANGE event for SchoolInfo for the 'From' Campus. b) Agent detects the change for the new campus and sends an ADD event for SchoolInfo (including Entity_Open date If the close of a campus effects the AdminStatus or ParentSchool of another campus then a CHANGE event for the effected Campus(es) also needs to be published.

Object Behaviour

Object	Message	Behaviour	Choreography
SchoolInfo	Event	Change	Entity_Close date published from closing campus
SchoolInfo	Event	Add	Entity_Open date must be published for transferred campus

SchoolInfo

Refer to TABLE S1.0

Change: Elements in this table to be supplied with a "Edit/Change" when the elements change together with the following identifiers required for consistency.

Add: Mandatory elements in TABLE S1.0 to be supplied with an "Add" School/Campus

Element/Attribute	Char	Baseline Char	Description	Business Rules
RefId	M	M	The GUID that identifies this school/campus	
LocalId	M	M	The locally-assigned identifier for this school/campus	

	StateProvinceId	O	S	The state-assigned identifier for this school/campus	
	CommonwealthId	O	S	Commonwealth identifier for this school/campus.	
	ACARAId	O	S	ACARA identifier for this school/campus.	
	SchoolName	M	M	Name of school/campus.	
	Campus (and associated elements)	O	C	Campus identifier	Must be published in a split of a campus if any change is detected in sub-elements.
	Campus/AdminStatus	M	M	Is this Campus the Administration Campus?	Must be published in a split of a campus if any change are detected.
	Campus/CampusType	O	O	Type of campus.	Must be published in a split of a campus if any change are detected.
	Campus/ParentSchoolId	C	CS	Parent School Identifier	Parent School Id needs to be supplied if this Campus is not the Admin Campus. This number should be the state-assigned identifier for this school or the locally-assigned identifier for this school. Must be published in a split of a campus if any change are detected Only available from 1.3 onwards.

3.1 Student Scenarios

2. Future Enrollment
3. Future --> Current Enrollment
4. New Current Enrollment
5. Year End Rollover
6. Exit School
7. Change Campus
8. Mid-Year Promotion/Demotion
9. Change in Personal Details
10. Census Reporting

3.1.2 Future Enrollment

When a student is identified as likely to attend in an upcoming academic year the SIS, where it is the SoE, shall generate events accordingly:

Pre-requisite:

SIF: SchoolInfo previously published to the zone

Event	Description - Trigger	User Input	Agent Action - Publisher
Future Enrollment	New student added for a future date.	Jurisdiction staff adds a new student to the SIS.	Agent detects the change and sends an ADD event for StudentPersonal and StudentSchoolEnrollment with "F" for TimeFrame.

Object	Message	Behaviour	Choreography
StudentPersonal	Event	Create	Event to be published by SIS. It is recommended, (but not required), that the event should be published in near real-time
StudentSchoolEnrollment	Event	Create	Event to be published by SIS. It is recommended (but not required) that the event should be published in near real-time. This event is to be published after the StudentPersonal event. Should multiple students be affected by a transaction in the SIS, it is an implementation decision whether all StudentPersonal events are published, then followed by all SSEs; or a sequence of several SP/SSE pairs are published.

Data Elements that need to be sent as a minimum – other 'O' optional elements in the full SIF Specification can be published, but as a minimum subscriber's must be able to consume the following:

(Note: elements in blue below have been added to StudentPersonal for Jurisdictions' provisioning for National Reporting purposes such as School Census. They are not required or suggested to provision on-line systems or applications.)

Student Personal

Attribute /Key	Element/Attribute	Char	Baseline Char	Description	Business Rules
Identifier					
@	RefId	M	M	The GUID of the student.	
	LocalId	M	M	The locally-assigned identifier for this student.	
	StateProvinceId	O	S	The state-assigned identifier for this student.	Provide if this is available.
	OtherIdList	O	O	Lists all "other" identifiers	

				associated with the student	
	OtheridList/Otherid	O	MR	Lists an "other" identifier associated with the student	
	PersonInfo/Name/FamilyName	C	M	Family name. That part of the person's name which is used to describe family, clan, tribal group, or marital association.	
	PersonInfo/Name/GivenName	C	M	Given name of the person.	
	PersonInfo/Name/MiddleName	C	S	All other given or middle names, each separated with a single space character.	
	PersonInfo/Name/PreferredGivenName	O	S	The given name preferred most by the person (as written).	If you don't store preferred name then provide GivenName
	PersonallInfo/AddressList/Street/StreetNumber	O	S	The address number assigned to building	
	PersonallInfo/AddressList/Street/StreetName	O	S	The name of the street	
	PersonallInfo/AddressList/City	O	S	The city part of the address	
	PersonallInfo/AddressList/StateProvince	C	S	The state or province code	
	PersonallInfo/AddressList/PostalCode	M	S	The ZIP/postal code	
	PersonallInfo/AddressList/GridLocation/Longitude	O	O	The longitude of street address	
	PersonallInfo/AddressList/GridLocation/Latitude	O	O	The latitude of the street address	
	PersonInfo/EmailList	O	S	The person's e-mail address(es).	
	PersonInfo/EmailList/Email	O	S	The Primary e-mail address associated with an individual or organization. (as per business rule)	If this attribute value is captured and available within the SIS it is suggested that it is published wherever possible. It is recognised that not all SIS make this attribute mandatory, nor do school processes enforce the capture.
@	PersonInfo/EmailList/Email/Type	M	M	Type of email address.	'01' Email Address provided must be the Primary Email address.
	MostRecent/YearLevel	O	M	The current or most recent grade level of the student in the district.	
	MostRecent/Parent1Language	O	S	Main Language Spoken at home by Parent 1, other than English	

Commented [NN1]: Line1 is mandatory

	MostRecent/Parent2Language	O	S	Main Language Spoken at home by Parent 2, other than English	
	MostRecent/Parent1SchoolEducationLevel	O	S	The highest level of school education completed by Parent1	
	MostRecent/Parent2SchoolEducationLevel	O	S	The highest level of school education completed by Parent2	
	MostRecent/Parent1NonSchoolEducation	O	S	The highest level of non-school education completed by Parent1	
	MostRecent/Parent2NonSchoolEducation	O	S	The highest level of non-school education completed by Parent2	
	MostRecent/Parent1EmploymentType	O	S	Code that defines employment type. This is compulsory for Parents or Guardians of the Student.	0-9 codes
	MostRecent/Parent2EmploymentType	O	S	Code that defines employment type. This is compulsory for Parents or Guardians of the Student.	0-9 codes
	PersonInfo/Demographics/Sex	O	S	'Sex' is the distinction 'male' and 'female', as reported by the person	
	PersonInfo/Demographics/BirthDate	O	S	The person's date of birth.	
	PersonInfo/Demographics/IndigenousStatus	O	S	Whether or not the student identifies as being of Aboriginal or Torres Strait Islander descent	
	PersonInfo/Demographics/CountryofBirth	O	S	The person's country of birth	
	PersonInfo/Demographics/LanguageList	O	S	List of Languages the person uses to communicate	

StudentSchoolEnrollment

(Mandatory for Create)

The Alternate Key is identified in the tables below with 'AK'. It is the combination of all Alternate Keys identified that form the key. i.e. it is a composite alternative key, rather than separate individual keys. If any one of these Alternate Keys were to be 'Change' it would break the immutable AK rule. (The SIS would instead need to generate a new instance of a SSE object, and change the existing instance where appropriate).

Attribute/Key	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The ID (GUID) that uniquely identifies a particular enrollment.	
AK	StudentPersonalRefId	M	M	The ID (GUID) of the student to whom this information is linked.	
AK	SchoolInfoRefId	M	M	The ID (GUID) of the school to which this enrollment applies.	
	MembershipType	M	M	The type of this enrollment as it relates to the school identified in SchoolInfoRefId	
AK	SchoolYear	M	M	School year for which this enrollment is applicable, expressed as the four-digit year in which the school year ends (e.g. 2007). StudentSchoolEnrollment instances must not span multiple school years.	As this is a 'Future' enrolment, this value may be greater than the current SchoolYear
	TimeFrame	M	M	The timeframe of the enrollment based on the SIF_Date in the SIF_Header of the message. For events, it is determined as of the date the event is generated. For requests and responses, it is calculated based on the date of the request.	= 'F' for future
	EntryDate	M	M	The date from when this enrollment will be valid.	
	YearLevel	O	M	Expected Year level of student when their enrolment becomes 'Current'.	
	FTE	O	S	Full-time equivalent numeric value of the student's course load during this enrollment, expressed in decimal form, where 1.00 represents a full-time enrollment	

3.1.3 Future->Current Enrollment

This scenario covers the situation where a student already identified to the zone as a 'Future' student transitions into being a 'current' student. This transition will likely take place during year-end rollover processing. It is assumed that when the 'Future' SSE was created a value for 'SchoolYear' was used that is still appropriate to the Current enrolment. If this is not the case (i.e. the student deferred a year) then 'Future' SSE cannot be used as the 'SchoolYear' attribute forms part of the Alternate Key.

Event	Description - Trigger	User Input	Agent Action - Publisher
Future -> Current Enrollment	'Future' student transitions into being a 'current' student.	Manual update of an individual student or automated processing such as EOY.	Agent detects the change and sends a CHANGE event for StudentSchoolEnrollment.

Object	Message	Behaviour	Notes

StudentSchoolEnrollment	Event	Change	Both StudentPersonal and SchoolInfo must already have been published to the zone by the designated 'provider'
-------------------------	-------	--------	---

StudentSchoolEnrollment

The primary key of this Object is the RefId and the minimum no of elements that need to be sent to change a student from "Future" to "Current" is the TimeFrame element. Other elements that have changed need also to be sent with the Change event. Elements that cannot change during the lifetime of this object (the 'Alternate Key') are identified in the "Future" and "Current" Enrolment Scenarios.

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The ID (GUID) a particular student	
	TimeFrame	M	M	The timeframe of the enrollment based on the SIF_Date in the SIF_Header of the message. For events, it is determined as of the date the event is generated. For requests and responses, it is calculated based on the date of the request.	= 'C' for Current The TimeFrame value changes when a student becomes current.

3.1.4 New Current Enrollment

Event	Description - Trigger	User Input	Agent Action - Publisher
Add new current enrollment	New current student is added	School Staff add a new current student to an SIS.	Agent detects the change and sends an ADD event for StudentSchoolEnrollment (and possibly StudentPersonal and SchoolInfo if they don't already exist in the Zone).

Object	Message	Behaviour	Notes
StudentPersonal	Event	Create	StudentPersonal to be published by the designated 'provider' if not previously published in the zone. Event to be published by SIS. It is recommended, (but not required), that the event should be published in near real-time
SchoolInfo	Event	Create	SchoolInfo to be published to the zone by the designated 'provider' if not previously published.
StudentSchoolEnrollment	Event	Create	Both StudentPersonal and SchoolInfo must already have been published to the zone by the designated 'provider'. This event is to be published after the StudentPersonal event. Should multiple students be affected by a transaction in the SIS, it is a implementation decision whether all StudentPersonal events are published, then followed by all SSEs; or a sequence of several SP/SSE pairs are published.

StudentSchoolEnrollment

The immutable Alternative Key elements are identified with an 'AK' key identifier.

Attribute/ Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The ID (GUID) that uniquely identifies a particular enrollment.	
AK	StudentPersonalRefId	M	M	The ID (GUID) of the student to whom this information is linked.	
AK	SchoolInfoRefId	M	M	The ID (GUID) of the school to which this enrollment applies.	
	MembershipType	M	M	The type of this enrollment as it relates to the school identified in SchoolInfoRefId	
AK	SchoolYear	M	M	School year for which this enrollment is applicable, expressed as the four-digit year in which the school year ends (e.g. 2007). StudentSchoolEnrollment instances must not span multiple school years.	As this is a "Current" enrolment, this value must be equal to the current SchoolYear
	TimeFrame	M	M	The timeframe of the enrollment based on the SIF_Date in the SIF_Header of the message. For events, it is determined as of the date the event is generated. For requests and responses, it is calculated based on the date of the request.	= 'C' for Current
	EntryDate	M	M	The date from when this enrollment will be valid.	As this enrolment is current the EntryDate must be the same or less than the day of entry of the information into the SIS
	YearLevel	O	M	Year level or academic level of student for this enrolment.	

3.1.5 Year End Rollover

Event	Description - Trigger	User Input	Agent Action - Publisher
Year End Rollover	End of a School Year	School Staff initiate EOY processes.	Agent detects the change and sends a CHANGE event for StudentSchoolEnrollment old school year and an ADD event for StudentSchoolEnrollment for the new school year.

AcademicYear of SSE is a primary key. Therefore the Enrollment must now refer to a new key, and following the immutable primary key rule, this means a different SSE instance must be used

Object	Message	Behaviour	Choreography
StudentSchoolEnrollment	Event	Change	It is recommended that the 'Change' SSE event is published prior to the new enrolment
StudentSchoolEnrollment	Event	Create	It is recommended this event is published after the 'Change' SSE event. This can either be in batches or pairs.

Recommendation:

In some cases, the SIS may provide the capability for schools to configure 'next ' year whilst the 'current' year is still active. It is recommended that these 'future' configurations are *not* issued as SIF messages. Furthermore if a SIS provides the functionality for schools to 'try out' next year 'just to see how it looks' that those changes are not published to SIF. This may mean the SIS needs to implement a form of 'formally commit next year' that triggers the SIF events that were previously not published.

StudentSchoolEnrollment - Change

At Year End Rollover, the current enrolment is made historical and a new StudentSchoolEnrollment is compiled and sent.

Attribute/ Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The Id (GUID) a particular student	
	TimeFrame	M	M	The timeframe of the enrollment based on the SIF_Date in the SIF_Header of the message.	= 'H' for Historical The TimeFrame value changes when a student enrolment is closed.
	ExitDate	C	M	The last school calendar day of this enrollment.	
	RecordClosureReason	O	M	The reason why this enrollment was closed. The EndOfYear option must be used to convey status change due to rollover activity.	= EndOfYear

StudentSchoolEnrollment - Create

The immutable Alternative Key elements are identified with an 'AK' key identifier.

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
--------------------------	-------------------	------	---------------	-------------	----------------

@	RefId	M	M	The ID (GUID) a particular student	
AK	StudentPersonalRefId	M	M	The ID (GUID) of the student to whom this information is linked.	
AK	SchoolInfoRefId	M	M	The ID (GUID) of the school to which this enrollment applies.	
	MembershipType	M	M	The type of this enrollment as it relates to the school identified in SchoolInfoRefId	
AK	SchoolYear	M	M	School year for which this enrollment is applicable, expressed as the four-digit year in which the school year ends (e.g. 2007). StudentSchoolEnrollment instances must not span multiple school years.	As this is a "Current" enrolment, this value must be equal to the current SchoolYear
	TimeFrame	M	M	The timeframe of the enrollment based on the SIF_Date in the SIF_Header of the message. For events, it is determined as of the date the event is generated. For requests and responses, it is calculated based on the date of the request.	= 'C' for Current
	EntryDate	M	M	The date from when this enrollment will be valid.	As this enrolment is current the EntryDate must be the same or less than the day of entry of the information into the SIS
	YearLevel	O	M	Year level or academic level of student for this enrolment.	

TeachingGroup

At Year End Rollover, it is assumed that there will be a new set of TeachingGroup Objects generated through the normal process of set up of a new SchoolYear. Where there are bulk changes to be made to the StudentList, TeacherList or TeachingGroupPeriodList - all changes should be made in the one change event where feasible to prevent excessive change events being published.

3.1.6 Exit School

Event	Description - Trigger	User Input	Agent Action - Publisher
-------	-----------------------	------------	--------------------------

Exit School	Student leaves a school	School Staff start student the Student exit process in a SIS.	Agent detects the change and sends an CHANGE event for StudentSchoolEnrollment updating exit columns as below.
--------------------	-------------------------	---	--

Object	Message	Behaviour	Choreography
StudentSchoolEnrollment	Event	Change	Event to be published by SIS. It is recommended (but not required) that the event should be published in near real-time.

StudentSchoolEnrollment

<i>Attribute/Key Identifier</i>	<i>Element/Attribute</i>	<i>Char</i>	<i>Baseline Char</i>	<i>Description</i>	<i>Business Rules</i>
@	Refid	M	M	The ID (GUID) a particular student	
	TimeFrame	M	M	The timeframe of the enrollment based on the SIF_Date in the SIF_Header of the message.	= 'H' for Historical The TimeFrame value changes when a student enrolment is closed.
	ExitDate	C	M	The last school calendar day of this enrollment.	
	ExitStatus/Code	O	M	Code indicating the closure status for this enrolment.	Choose the appropriate exit type from AU Codeset.
	RecordClosureReason	O	M	The reason why this enrollment was closed. The EndOfYear option must be used to convey status change due to rollover activity.	= SchoolExit

3.1.7 Change Campus

A Campus is a distinct SchoolInfo object instance in SIF. Therefore the Enrollment must now refer to a new Foreign key, and following the immutable primary key rule, this means a different SSE instance must be used. In essence, 'Change Campus' is similar to an 'Exit School' + 'New Enrollment'.

Event	Description - Trigger	User Input	Agent Action - Publisher
Change	A student changes campus in the same	School Staff change the Campus of a	Agent detects the change and sends a CHANGE event for

Campus	School.	Student.	StudentSchoolEnrollment for the 'old' campus and an ADD event for StudentSchoolEnrollment (in the new Campus possibly SchoolInfo if it doesn't already exist in the Zone).
---------------	---------	----------	--

Object	Message	Behaviour	Choreography
StudentSchoolEnrollment	Event	Change (for existing object)	Event to be published by SIS. It is recommended (but not required) that the event should be published in near real-time.
StudentSchoolEnrollment	Event	Create (for new object)	Event to be published by SIS. It is recommended (but not required) that the event should be published in near real-time.

StudentSchoolEnrollment (existing) - Change

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The ID (GUID) a particular student	
	TimeFrame	M	M	The timeframe of the enrollment based on the SIF_Date in the SIF_Header of the message.	= 'H' for Historical The TimeFrame value changes when a student enrolment is closed.
	ExitDate	C	M	The last school calendar day of this enrollment.	
	RecordClosureReason	O	M	The reason why this enrollment was closed.	Options for RecordClosureReason: a. 'SchoolExit' - whilst the student hasn't technically left the school, they have left the 'SchoolInfo' object. b. 'CampusExit' - This is semantically more accurate. However, there may be situations where the system generating the exit does not know whether it is really a Campus change. (Lower school uses SIS from vendor A, whereas upper school uses SIS from vendor B.)

StudentSchoolEnrollment – Create

Attribute/	Element/Attribute	Char	Baseline	Description	Business Rules
------------	-------------------	------	----------	-------------	----------------

Key Identifier			Char		
@	RefId	M	M	The ID (GUID) a particular student	
AK	StudentPersonalRefId	M	M	The ID (GUID) of the student to whom this information is linked.	
AK	SchoolInfoRefId	M	M	The ID (GUID) of the school to which this enrollment applies.	
	MembershipType	M	M	The type of this enrollment as it relates to the school identified in SchoolInfoRefId	
AK	SchoolYear	M	M	School year for which this enrollment is applicable, expressed as the four-digit year in which the school year ends (e.g. 2007). StudentSchoolEnrollment instances must not span multiple school years.	As this is a "Current" enrolment, this value must be equal to the current SchoolYear
	TimeFrame	M	M	The timeframe of the enrollment based on the SIF_Date in the SIF_Header of the message. For events, it is determined as of the date the event is generated. For requests and responses, it is calculated based on the date of the request.	= 'C' forCurrent
	EntryDate	M	M	The date from when this enrollment will be valid.	As this enrolment is current the EntryDate must be the same or less than the day of entry of the information into the SIS
	YearLevel	O	M	Year level or academic level of student for this enrolment.	

3.1.8 Mid-year Promotion/Demotion

Event	Description - Trigger	User Input	Agent Action - Publisher
Mid-year Promotion/Demotion	Student has a change in Year level during a School Year.	School Staff update a student's school year in StudentSchoolEnrollment and MostRecent container in StudentPersonal if previously published.	Agent detects the change and sends an CHANGE event for StudentSchoolEnrollment (and possibly StudentPersonal MostRecent container).

Object	Message	Behaviour	Choreography
StudentSchoolEnrollment	Event	Change	For TimeFrame = 'C'; YearLevel will change, PromotionInfo MAY be Supplied
StudentPersonal	Event	Change	MostRecent/YearLevel will be updated with new Year if published in the Zone.

StudentSchoolEnrollment – Change

Attribute/ Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The ID (GUID) a particular student	
AK	StudentPersonalRefId	M	M	The ID (GUID) of the student to whom this information is linked.	
AK	SchoolInfoRefId	M	M	The ID (GUID) of the school to which this enrollment applies.	
	TimeFrame	M	M	The timeframe of the enrollment based on the SIF_Date in the SIF_Header of the message. For events, it is determined as of the date the event is generated. For requests and responses, it is calculated based on the date of the request.	= 'C' forCurrent
	YearLevel	O	M	Year level or academic level of student for this enrolment.	Modified to reflect 'Promotion/Demotion'
	PromotionInfo	O	O	Information on a student's promotion, retention, or demotion related to this enrollment instance.	This information is optional for the both the Specification and the Baseline Profile.
	PromotionInfo/PromotionStatus	O	O	This value should be set if this enrollment instance is closed for an end-of-year closeout or a mid-year promotion or demotion. A value other than NA should be specified when the enrollment instance represents an end-of-year closeout or mid-year promotion/demotion.	values: to be used for Mid-year Promotion or Demotion. Promoted, Demoted

StudentPersonal– Change

Student Personal

Attribute /Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules

@	RefId	M	M	The GUID of the student.
	LocalId	M	M	The locally-assigned identifier for this student.
	MostRecent/YearLevel	O	M	The current or most recent grade level of the student in the district.

3.1.9 Change in Personal Details

Event	Description - Trigger	User Input	Agent Action - Publisher
Change in Personal Details	Student has a change in personal details.	School Staff update information about a student.	Agent detects the change and sends an CHANGE event for StudentPersonal.

Summary: This scenario caters for a change in any personal details of the student that does not impact their StudentSchoolEnrollment e.g. name change, contact details change.

Pre-Requisites: The student is already known to other systems.

Object	Message	Behaviour	Choreography
StudentPersonal	Event	Change	Event to be published by SIS. It is recommended (but not required) that the event should be published in near real-time

Only the information added or changed together with the Primary Key (RefId) needs to be re-published.

Student Personal

Attribute /Key	Element/Attribute	Char	Baseline Char	Description	Business Rules
Identifier					
@	RefId	M	M	The GUID of the student.	
	LocalId	M	M	The locally-assigned identifier for this student.	
	StateProvinceId	O	S	The state-assigned identifier for this student.	Provide if this is available.
	OtherIdList	O	O	Lists all "other" identifiers associated with the student	

	OtheridList/Otherid	O	MR	Lists an "other" identifier associated with the student	
	PersonInfo/Name/FamilyName	C	M	Family name. That part of the person's name which is used to describe family, clan, tribal group, or marital association.	
	PersonInfo/Name/GivenName	C	M	Given name of the person.	
	PersonInfo/Name/MiddleName	C	S	All other given or middle names, each separated with a single space character.	
	PersonInfo/Name/PreferredGivenName	O	S	The given name preferred most by the person (as written).	If you don't store preferred name then provide GivenName
	PersonallInfo/AddressList/Street/StreetNumber	O	S	The address number assigned to building	
	PersonallInfo/AddressList/Street/StreetName	O	S	The name of the street	
	PersonallInfo/AddressList/City	O	S	The city part of the address	
	PersonallInfo/AddressList/StateProvince	C	S	The state or province code	
	PersonallInfo/AddressList/PostalCode	M	S	The ZIP/postal code	
	PersonallInfo/AddressList/GridLocation/Longitude	O	O	The longitude of street address	
	PersonallInfo/AddressList/GridLocation/Latitude	O	O	The latitude of the street address	
	PersonInfo/EmailList	O	S	The person's e-mail address(es).	
	PersonInfo/EmailList/Email	O	S	The Primary e-mail address associated with an individual or organization. (as per business rule)	If this attribute value is captured and available within the SIS it is suggested that it is published wherever possible. It is recognised that not all SIS make this attribute mandatory, nor do school processes enforce the capture.
@	PersonInfo/EmailList/Email/Type	M	M	Type of email address.	= '01' Email Address provided must be the Primary Email address.
	MostRecent/YearLevel	O	M	The current or most recent grade level of the student in the district.	
	MostRecent/Parent1Language	O	S	Main Language Spoken at home by Parent 1, other than English	
	MostRecent/Parent2Language	O	S	Main Language Spoken at home by Parent 2, other than English	

	MostRecent/Parent1SchoolEducationLevel	O	S	The highest level of school education completed by Parent1	
	MostRecent/Parent2SchoolEducationLevel	O	S	The highest level of school education completed by Parent2	
	MostRecent/Parent1NonSchoolEducation	O	S	The highest level of non-school education completed by Parent1	
	MostRecent/Parent2NonSchoolEducation	O	S	The highest level of non-school education completed by Parent2	
	MostRecent/Parent1EmploymentType	O	S	Code that defines employment type. This is compulsory for Parents or Guardians of the Student.	0-9 codes
	MostRecent/Parent2EmploymentType	O	S	Code that defines employment type. This is compulsory for Parents or Guardians of the Student.	0-9 codes
	PersonInfo/Demographics/Sex	O	S	'Sex' is the distinction 'male' and 'female', as reported by the person	
	PersonInfo/Demographics/BirthDate	O	S	The person's date of birth.	
	PersonInfo/Demographics/IndigenousStatus	O	S	Whether or not the student identifies as being of Aboriginal or Torres Strait Islander descent	
	PersonInfo/Demographics/CountryofBirth	O	S	The person's country of birth	
	PersonInfo/Demographics/LanguageList	O	S	List of Languages the person uses to communicate	

3.1.10 Census Reporting:

Clarification on how to publish Census data.

Demographic information intended for Census reporting will be added to Demographic information into the standard jurisdiction SBP model for Parents and Students.

To enable a simple and scalable model, where an Agent does not publish parent information (StudentContactPersonal) or where it is not available, in addition to the standard SBP model, Parent demographic information can be published in the StudentPersonal Object.

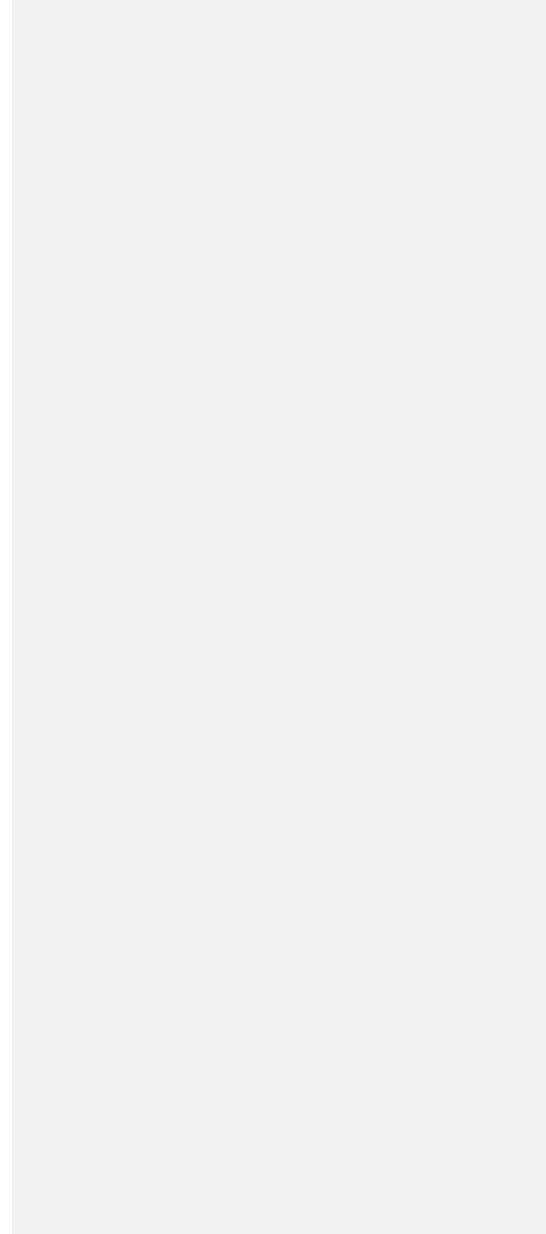
This allows the each implementation the choice of undertaking a census using only the StudentPersonal Object. However if the implementation is using StudentContactPersonal then the same parent information should be published there. This simplifies the model and enables all participating vendors to publish Census information simply, and a consistent path forward if parent entries are to published in the future.

StudentPersonal Census elements:

(refer to full StudentPersonal information above).

	MostRecent/Parent1Language	O	S	Main Language Spoken at home by Parent 1, other than English	
	MostRecent/Parent2Language	O	S	Main Language Spoken at home by Parent 2, other than English	
	MostRecent/Parent1SchoolEducationLevel	O	S	The highest level of school education completed by Parent1	
	MostRecent/Parent2SchoolEducationLevel	O	S	The highest level of school education completed by Parent2	
	MostRecent/Parent1NonSchoolEducation	O	S	The highest level of non-school education completed by Parent1	
	MostRecent/Parent2NonSchoolEducation	O	S	The highest level of non-school education completed by Parent2	
	MostRecent/Parent1EmploymentType	O	S	Code that defines employment type. This is compulsory for Parents or Guardians of the Student.	0-9 codes
	MostRecent/Parent2EmploymentType	O	S	Code that defines employment type. This is compulsory for Parents or Guardians of the Student.	0-9 codes
	PersonInfo/Demographics/IndigenousStatus	O	S	Whether or not the student identifies as being of Aboriginal or Torres Strait Islander descent	
	PersonInfo/Demographics/CountryofBirth	O	S	The person's country of birth	
	PersonInfo/Demographics/LanguageList	O	S	List of Languages the person uses to communicate	

DRAFT until approved by DSWG



3.2. Staff and Staff Assignment Scenarios

1. New Staff Member and School Assignment(s)
2. Additional School Assignment (existing Staff Member)
3. Change in Role (e.g. Promotion)
4. Staff Member leaves School
5. Staff Member goes on leave (extended)
6. Staff Member leaves Organisation
7. Change in Personal Details

3.2.1 New Staff Member and School Assignment(s)

Summary: This scenario addresses the situation when a new person joins the organisation as an employee. In addition, they may be optionally assigned to a school or schools.

Pre-Requisites: School data shared. Schools are known to other systems

Event	Description - Trigger	User Input	Agent Action - Publisher
New Staff Member and School Assignment(s)	New Staff joins an organisation.	School admin updates information about a new staff member in the SIS.	Agent detects the change and sends an ADD event for StaffPersonal and StaffAssignment/s.

Objects Events:

Object	Message	Behaviour	Choreography
StaffPersonal	Event	Create	It is recommended (but not a requirement) that the event is published in near real-time. StaffPersonal is sent before StaffAssignment. Multiple StaffAssignment messages can be sent for one StaffPersonalRefId. For events resulting in many new staff members, it is an implementation decision as to whether StaffPersonal and StaffAssignment be paired or all StaffPersonal are sent followed by all StaffAssignment objects.
StaffAssignment	Event	Create	For events resulting in many new staff members, it is an implementation decision as to whether StaffPersonal and StaffAssignment be paired or all StaffPersonal are sent followed by all StaffAssignment objects.

Object Attributes:

3.2.1.1 StaffPersonal

Attribute / Key Identifier	Element / Attribute	Char	Baseline Char	Description	Business Rules
	RefId	M	M	The GUID of the staff member.	
	LocalId	M	M	The locally-assigned identifier for this staff member.	
	StateProvinceId	O	S	The state-assigned identifier for this staff member.	
	OtherIdList	O	O	Lists all "other" identifiers associated with the staff member	
	OtherIdList/OtherId	O	MR	Lists an "other" identifier associated with the staff member	
	PersonInfo/Name/FamilyName	M	M		
	PersonInfo/Name/GivenName	M	M		
	PersonInfo/Name/MiddleName	C	S	All other given or middle names, each separated with a single space character.	
	PersonInfo/Name/PreferredGivenName	O	S		
	PersonInfo/EmailList/Email	O	S		
	PersonInfo/EmailList/Email/Type	M	M		= '01' Email provided must be Primary Email address.
	PersonInfo/Demographics/Sex	O	M		
	PersonInfo/PhoneNumberList/PhoneNumber/Number	O	S		
	PersonInfo/PhoneNumberList/PhoneNumber/Type	O	S		
	EmploymentStatus	O	S	Flags whether the staff member is currently employed by the organization (Active), no longer/not yet employed, or unavailable (on extended leave)	Should be set to "Active"

3.2.1.2 StaffAssignment

Attribute / Key Identifier	Element / Attribute	Char	Baseline Char	Description	Business Rules
	RefId	M	M	The GUID that uniquely identifies a particular staff assignment.	
	SchoolInfoRefId	M	M	The ID (GUID) that identifies the school where the staff member is assigned.	
	SchoolYear	M	M	School year for which the information is applicable, expressed as the four-digit year in which the school year ends (e.g., 2007). Release 1.2 allows this to be optional as JobStart and EndDate allow for this Object to be used over more than one year.	
	StaffPersonalRefId	M	M	ID (GUID) of this staff member, as represented in the StaffPersonal object.	
	PrimaryAssignment	M	M	Is this the staff member's primary assignment? Note: There must be one and only one instance of the object with a Yes value in this element per school year.	
	JobStartDate	O	S	This is the date from which the staff assignment is valid (inclusive).	
	JobEndDate	O	S	This is the date through which the staff assignment is valid (inclusive).	
	JobFunction	O	S	The purpose of the activities as related to students.	
	StaffActivity	O	M	The purpose of the Staff Member's role as related to students. Equivalent to 'JobFunction' in US Object.	
	StaffActivity/Code	M	M	Code representing the type of staff activity undertaken.	

3.2.2 Additional School Assignment (existing Staff Member)

Summary: This scenario addresses the situation when an existing staff member is assigned to a new school. It also may involve the end of tenure at another school as is often the case with transfers.

Pre-Requisites: The staff member is already known to other systems. School data shared. Schools are known to other systems. Staff member is available for assignment (proposed StaffMember/EmploymentStatus is Active)

Event	Description - Trigger	User Input	Agent Action - Publisher
Additional School Assignment	Staff gains an additional Assignment	School admin update information about a staff member's assignment.	Agent detects the change and sends a CHANGE event for StaffAssignment if this is an additional Assignment and an ADD event for the new StaffAssignment.

Objects Events:

Object	Message	Behaviour	Choreography
StaffAssignment	Event	Change	Update JobEndDate if previous tenure completed and update PrimaryAssignment if no longer the primary role.
StaffAssignment	Event	Create	New School Assignment

Object Attributes:**3.2.2.1 StaffAssignment – Change**

Attribute / Key Identifier	Element / Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The GUID that uniquely identifies a particular staff assignment.	
	SchoolInfoRefId	M	M	The ID (GUID) that identifies the school where the staff member is assigned.	
	SchoolYear	M	M	School year for which the information is applicable, expressed as the four-digit year in which the school year ends (e.g., 2007).	
	StaffPersonalRefId	M	M	ID (GUID) of this staff member, as represented in the StaffPersonal object.	
	PrimaryAssignment	M	M	Is this the staff member's primary assignment? Note: There must be one and only one instance of the object with a Yes value in this element per school year.	Update if this is no longer the primary assignment
	JobStartDate	O	S	This is the date from which the staff assignment is valid (inclusive).	
	JobEndDate	O	C	This is the date through which the staff assignment is valid (inclusive). (This needs to be provided if the change is an end of tenure.)	Where tenure is ended, a date must be provided to signal the completion of the StaffActivity at the school
	JobFunction	O	S	The purpose of the activities as related to students.	
	StaffActivity	O	M	The purpose of the Teacher's role as related to students. Equivalent to 'JobFunction' in US Object.	
	StaffActivity/Code	M	M	Code representing the type of staff activity undertaken.	

3.2.2.2 StaffAssignment - Create

Attribute / Key Identifier	Element / Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The GUID that uniquely identifies a particular staff assignment.	
	SchoolInfoRefId	M	M	The ID (GUID) that identifies the school where the staff member is assigned.	
	SchoolYear	M	M	The ID (GUID) that identifies the school where the staff member is assigned.	
	StaffPersonalRefId	M	M	ID (GUID) of this staff member, as represented in the StaffPersonal object.	
	PrimaryAssignment	M	M	Is this the staff member's primary assignment? Note: There must be one and only one instance of the object with a Yes value in this element per school year.	
	JobStartDate	O	S	This is the date from which the staff assignment is valid (inclusive).	
	JobEndDate	O	S	This is the date through which the staff assignment is valid (inclusive).	
	JobFunction	O	S	The purpose of the activities as related to students.	
	StaffActivity	O	M	The purpose of the Teacher's role as related to students. Equivalent to 'JobFunction' in US Object.	
	StaffActivity/Code	M	M	Code representing the type of staff activity undertaken.	

3.2.3 Change in Role (e.g. Promotion)

Summary: This scenario addresses the situation when an existing staff member is assigned a new role within an existing school. Their current role may continue (in parallel) or it may need to be ended.

Pre-Requisites: The staff member is already known to other systems. StaffPersonal has been sent. School data shared. Schools are known to other systems. Similar to 3.2.2.1 and 3.2.2.2 above.

Event	Description - Trigger	User Input	Agent Action - Publisher
Change in Role (e.g. Promotion)	Staff member is assigned a new role within an existing school.	School admin update information about a staff member's assignment.	Agent detects the change and sends a CHANGE event for StaffAssignment. For a new role where there is an additional Assignment agent sends an ADD event for the new StaffAssignment.

Objects Events:

Object	Message	Behaviour	Choreography
StaffAssignment	Event	Change – to end a role	Update JobEndDate to terminate role where required and update PrimaryAssignment if required
StaffAssignment	Event	Create – to add a new role	

Note: A request/response call where on a query for Job_End_Date > 0 on a StaffAssignment should never be sent and is not expected to be supported. (This means that a request for a StaffAssignment that has ended is not expected to be supported.)

Object Attributes:**3.2.3.1 StaffAssignment - Change**

Attribute/ Key Identifier	Element / Attribute	Char	Baseline Char	Description	Business Rules
	RefId	M	M		
	SchoolInfoRefId	M	M		
	SchoolYear	M	M		
	StaffPersonalRefId	M	M		
	JobStartDate	O	S		
	JobEndDate	O	M		Mandatory when ending a role
	PrimaryAssignment	M	M		Only one assignment can be set as the primary assignment. The existing primary assignment must first be set to no before the new primary assignment is set.
	JobFunction	O	S		
	StaffActivity/Code	M	M		

3.2.3.2 StaffAssignment - Create

Attribute/ Key	Element / Attribute	Char	Baseline Char	Description	Business Rules
-------------------	---------------------	------	------------------	-------------	----------------

Identifier					
	RefId	M	M		
	SchoolInfoRefId	M	M		
	SchoolYear	M	M		
	StaffPersonalRefId	M	M		
	PrimaryAssignment	M	M		Only one assignment can be set as the primary assignment. The existing primary assignment must first be set to no before the new primary assignment is set.
	JobStartDate	O	S		
	JobEndDate	O	S		
	JobFunction	O	S		
	StaffActivity/Code	M	M		

3.2.4 Staff Member leaves School

Summary: This scenario addresses the situation when a staff member finishes work at a school but remains an employee of the organisation. It is similar to "Additional School Assignment" above but does not involve a new assignment at another school.

Pre-Requisites: The staff member is already known to other systems. StaffPersonal has been sent. School data shared. Schools are known to other systems. Staff member is not leaving the organisation (proposed StaffPersonal/EmploymentStatus remains Active).

Event	Description - Trigger	User Input	Agent Action - Publisher
Staff Member leaves a School.	Staff finishes an assignment.	School admin update information about a staff member's assignment.	Agent detects the change and sends a CHANGE event for StaffAssignment updating the JobEndDate

Objects Events:

Object	Message	Behaviour	Choreography
StaffAssignment	Event	Change – to end a role	Update JobEndDate with date assignment ended

Object Attributes:

3.2.4.1 StaffAssignment - Change

Attribute/Key	Element / Attribute	Char	Baseline Char	Description	Business Rules
	RefId	M	M		
	SchoolInfoRefId	M	M		
	SchoolYear	M	M		
	StaffPersonalRefId	M	M		
	JobStartDate	O	S		
	JobEndDate	O	M		Mandatory when ending an Assignment.
	PrimaryAssignment	M	M		
	JobFunction	O	S		
	StaffActivity	O	M		
	StaffActivity/Code	M	M		

3.2.5 Staff Member goes on leave (extended)

Summary: This scenario addresses the situation when a staff member takes a leave of absence of an extended nature (such as long-service leave) and returns some time later.

Pre-Requisites: School data shared. Schools are known to other systems. The staff member is already known to other systems e.g. StaffPersonal has been sent.

Objects Events: StaffPersonal/EmploymentStatus was added to address this specific event in 1.2. Publishing an End Date and then re-opening the record on the Staff's return or issuing a new StaffAssignment was considered by the SBP group as not sufficient. A staff member can still be employed by a jurisdiction when on leave even though they may be ineligible to access resources. When a member is on extended leave but expected to return to the employment within an organisation, the resolution was to make a staff member 'temporarily inactive' by updating the EmploymentStatus indicating that they are on leave.

Event	Description - Trigger	User Input	Agent Action - Publisher
Staff Member goes on extended leave	Staff member goes on leave.	School admin update employment staus.	Agent detects the change and sends a CHANGE event for StaffPersonal. EmploymentStatus is updated to "on-leave".

Object	Message	Behaviour	Choreography
StaffPersonal	Event	Change	StaffPersonal must have already been published. StaffAssignment must have already been published (staff are unavailable for assignment while not Active). Employment Status to be set to "On-Leave"

Object Attributes:

3.2.5.1 StaffPersonal

Attribute/ Key Identifier	Element / Attribute	Char	Baseline Char	Description	Business Rules
	RefId	M	M		
	LocalID	M	M		
	StateProvinceID	O	S		
	PersonInfo/Name/FamilyName	C	M		
	PersonInfo/Name/GivenName	C	M		
	PersonInfo/Name/MiddleName	C	S	All other given or middle names, each separated with a single space character.	
	PersonInfo/Name/PreferredGivenName	O	S		
	PersonInfo/EmailList/Email	O	S		
	PersonInfo/EmailList/Email/Type	M	M		= '01' Email provided must be Primary Email address.
	PersonInfo/Demographics/Sex	O	M		
	PersonInfo/PhoneNumberList/PhoneNumber/Number	O	S		
	PersonInfo/PhoneNumberList/PhoneNumber/Type	O	S		
	EmploymentStatus	O	S	Flags whether the staff member is currently employed by the organization (active), no	Should be set to "OnLeave"

				longer/not yet employed, or unavailable (on extended leave)	
--	--	--	--	---	--

3.2.6 Staff Member leaves Organisation

Summary: This is the scenario where a staff member ends employment with the organisation and any assigned schools.

Pre-Requisites: School data shared. Schools are known to other systems. The staff member is already known to other systems.

Objects Events: - Previously, it was not clear how to deal with disabling a StaffPersonal record. What attribute can be used to identify a staff member leaving an organisation.? - The lack of Staff Assignments does not necessarily mean the staff member is inactive. StaffPersonal/EmploymentStatus has been added to address this. This element should be used and set to "Inactive" when a staff member leaves and organisation.

Event	Description - Trigger	User Input	Agent Action - Publisher
Staff Member leaves an Organisation	Staff member ends employment with organisation.	School admin update information about a staff member making them inactive.	Agent detects the change and sends a CHANGE event for StaffAssignment and StaffPersonal. StaffPersonal should not be deleted from the Zone without being set to Inactive for a reasonable amount of time to enable other applications to note that the Staff Member is not current. StaffAssignment JobEndDate to be updated with the End Date.

Object	Message	Behaviour	Choreography
StaffPersonal	Event	Change	Must be set to Inactive for a reasonable amount of time (so that other systems are notified of unavailability of staff) before StaffPersonal record is deleted from a zone.
StaffAssignment	Event	Change	Must be closed off before StaffPersonal is set to Inactive

Object Attributes:

3.2.6.1 StaffAssignment – Change Event

Attribute/ Key Identifier	Element / Attribute	Char	Baseline Char	Description	Business Rules
	RefId	M	M		
	SchoolInfoRefId	M	M		
	SchoolYear	M	M		
	StaffPersonalRefId	M	M		
	JobStartDate	O	S		
	JobEndDate	O	M		Mandatory when an Assignment is ended.
	PrimaryAssignment	M	M		
	JobFunction	O	S		
	StaffActivity	O	S		
	StaffActivity/Code	M	M		

3.2.6.2 StaffPersonal

Attribute / Key Identifier	Element / Attribute	Char	Baseline Char	Description	Business Rules
	RefId	M	M		
	LocalID	M	M		
	StateProvinceID	O	S		
	OtherIdList	O	O		
	OtherIdList/OtherId	O	MR		
	PersonInfo/Name/FamilyName	C	M		
	PersonInfo/Name/GivenName	C	M		SBP expects first name.
	PersonInfo/Name/MiddleName	C	S	All other given or middle names, each separated with a single space character.	
	PersonInfo/Name/PreferredGivenName	O	S		
	PersonInfo/EmailList/Email	O	O		

	PersonInfo/EmailList/Email/Type	M	M		= '01' Email provided must be Primary Email address.
	PersonInfo/Demographics/Sex	O	M		
	PersonInfo/PhoneNumberList/PhoneNumber/Number	O	O		
	PersonInfo/PhoneNumberList/PhoneNumber/Type	O	O		
	EmploymentStatus	O	S	Flags whether the staff member is currently employed by the organization (active), no longer/not yet employed, or unavailable (on extended leave)	Should be set to "Inactive"

3.2.7 Change in Personal Details

Summary: This scenario caters for a change in any personal details of the staff member that does not impact their staff assignment e.g. name change, contact details change.

Pre-Requisites: The staff member is already known to other systems.

Event	Description - Trigger	User Input	Agent Action - Publisher
Change in Staff Personal details	Staff member's personal details change.	School admin updates personal information about a staff member.	Agent detects the change and sends a CHANGE event for StaffPersonal updating any personal information that has changed.

Objects Events:

Object	Message	Behaviour	Choreography
StaffPersonal	Event	Change	Any personal details that change are published.

Object Attributes:

3.2.7.1 StaffPersonal

Attribute/ Key Identifier	Element / Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The GUID of the staff member.	
	LocalID	M	M	The locally-assigned identifier for this staff member.	
	StateProvinceID	O	S	The state-assigned identifier for this staff member.	
	OtherIdList	O	O		
	OtherIdList/OtherId	O	MR		
	PersonInfo/Name/FamilyName	C	M	Family name. That part of the person's name which is used to describe family, clan, tribal group, or marital association. Note that this element is required when known. However, it may not be possible to know the family name in which case you should add the available information to the FullName element.	
	PersonInfo/Name/GivenName	C	M	Given name of the person. Note that this element is required if known. However, it may not be possible to know the given name in which case you should add the available information to the FullName element.	
	PersonInfo/Name/MiddleName	C	S	All other given or middle names, each separated with a single space character.	
	PersonInfo/Name/PreferredGivenName	O	S	The given name preferred most by the person (as written).	
	PersonInfo/EmailList	O	S		EmailList is Optional, SBPGroup is strongly suggesting that at least one email address be sent.
	PersonInfo/EmailList/Email	M	M	This element represents an e-mail address of one of a number of types and occurs in objects such as StudentPersonal, StaffPersonal,	

				StudentContactPersonal, etc	
	PersonInfo/EmailList/Email/Type	M	M	This attribute specifies the type of e-mail address.	
	PersonInfo/Demographics/Sex	O	M	'Sex' is the distinction 'male' and 'female', as reported by the person	
	PersonInfo/PhoneNumberList/PhoneNumber/Number	O	S	Phone number. Free-form, but typical Australian formats include: •(###) ####-####, •####-####, •+##### ### ###	
	PersonInfo/PhoneNumberList/PhoneNumber/Type	O	S	Code that specifies what type of phone number this is. Note: A subset of valid values may be specified in data objects.	

DRAFT until approved by [unclear]

3.3. Use Cases for the StudentContactPersonal and StudentContactRelationship

3.3.1. Student Contacts in Profiles and Learning Areas:

We display the Student Contact information from the SISs in our web application. This brings all the information that teachers are looking for into one location and saves them having to learn another software application in order to retrieve Student Contact information. It also saves the school the trouble and cost of producing books of contacts to be stored in staff areas that quickly become out of date. As our application is web-based this approach allows us to make available the contact information from offsite via the website, something that would not be possible using the SIS's software without the use of yet another web-based application from the SIS if available. This makes the relevant student contacts available to teachers in the areas of our application that they spend most of their time working in without having to go to another application or location to retrieve contact information.

3.3.2. Student Attendance Follow-up:

When doing follow-up of unexplained absences, rather than having office staff continually flicking between screens to get contact information and enter the follow-up data, we display the student's contacts in the follow-up screen to make the process of chasing the absences as quick and easy as possible for staff tasked with this time consuming process.

3.3.3. SMS sending:

We have several 3rd party SMS gateway services that we can connect to and allow the sending of student absence SMSs and bulk SMSs to parents of students. In order to do this we need access to the mobile phone numbers of parents. Having the StudentContact objects as part of the baseline would enable us to keep our copy of these contacts synced and up to date, thereby eliminating the need of us having to maintain separate "lookup" modules to retrieve the information in real time from SIS DB directly.

3.3.4 StudentContact and StudentContactRelationship Provisioning

The Student Contact and Relationship consists of the following Scenarios:

1. Add a new StudentContact Person (a new contact and relationship)
2. Add a new StudentContactRelationship (to a current contact eg. sibling starts school)
3. Change a StudentContact Person information
4. Change a StudentContactRelationship (eg. changed access rights)
5. Remove a StudentContact Person from a Student

3.3.4.1 New Student Contact and Relationship

Summary: This scenario addresses the situation where a new student contact is added to a student for the first time. It assumes the Contact does not already exist in the SIS.

Pre-requisites: StudentPersonal already exists in the zone.

Event	Description - Trigger	User Input	Agent Action - Publisher
New Student Contact	New contact is added to a student	School admin creates a new contact and creates an association with a student.	Agent detects new contact and sends an ADD event for StudentContactPersonal and sends an ADD event for StudentContactRelationship.

Objects Events:

Object	Message	Behaviour	Choreography
StudentContactPersonal	Event	Create	Event to be published by SIS. It is recommended (but not required) that the event should be published in near real-time.
StudentContactRelationship	Event	Create	Event to be published by SIS. It is recommended (but not required) that the event should be published in near real-time.

3.3.4.1.1 StudentContactPersonal

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The ID (GUID) that uniquely identifies the instance of the object.	
	LocalId	O	O	The locally-assigned identifier for this person. Note: LocalId may be used as a unique person identifier rather than a unique contact identifier. In this case there may be more than one StudentContactPersonal	

				instance with the same LocalId.
	PersonInfo/Name/Title	O	O	A title or prefix associated with the name. If any of: Mr, Mrs, Ms, Miss, Rev, Fr, Dr, Prof, Hon, Sir, Lord, Lady - these must be as shown, otherwise free text. Note that title is not applicable to learners.
	PersonInfo/Name/FamilyName	C	C	Family name. That part of the person's name which is used to describe family, clan, tribal group, or marital association. Note that this element is required when known. However, it may not be possible to know the family name in which case you should add the available information to the FullName element.
	PersonInfo/Name/GivenName	C	S	Given name of the person. Note that this element is required if known.
	PersonInfo/Name/PreferredFamilyName	C	S	The family name preferred most by the person (as written).
	PersonInfo/Name/MiddleName	C	S	All other given or middle names, each separated with a single space character.
	PersonInfo/Name/PreferredGivenName	C	C	The given name preferred most by the person (as written).
	PersonInfo/Demographics/Sex	O	O	'Sex' is the distinction 'male' and 'female', as reported by the person
	PersonInfo/Demographics/LanguageList	O	O	List of languages an individual uses to communicate.
	PersonInfo/AddressList/Address/Type	M	M	Code that defines the location of the address. Note: A subset of specific valid values for each instance in a data object may be listed in that object.
	PersonInfo/AddressList/Address/Role	M	M	A facet of Address
	PersonInfo/AddressList/Address/Street/Line1	M	M	Address line 1.
	PersonInfo/AddressList/Address/Street/Line2	O	O	Address line 2.
	PersonInfo/AddressList/Address/City	M	M	The city part of the address.
	PersonInfo/AddressList/Address/StateProvince	C	C	The state or province code. StateTerritory Code only applicable if an Australian Address. Conditional- If Australian Address this is mandatory.
	PersonInfo/AddressList/Address/PostalCode	M	M	The ZIP/postal code.
	PersonInfo/AddressList/Address/Country	O	O	The country code.

	PersonInfo/PhoneNumberList/PhoneNumber/Type	M	M	Code that specifies what type of phone number this is. Note: A subset of valid values may be specified in data objects.	
	PersonInfo/PhoneNumberList/PhoneNumber/Number	M	M	Phone number. Free-form, but typical Australian formats include: (###) ##### ####, ##### ####, +##### ### ##	
	PersonInfo/EmailList	O	S		EmailList is Optional, SBPGroup is strongly suggesting that at least one email address be sent.
	PersonInfo/EmailList/Email/Type	M	M	This attribute specifies the type of e-mail address	
	PersonInfo/EmailList/Email	M	M	This element represents an e-mail address of one of a number of types and occurs in objects such as StudentPersonal, StaffPersonal, StudentContactPersonal, etc.	
	SchoolEducationalLevel	O	O	The highest level of school education completed by the contact person.	
	NonSchoolEducation	O	O	The highest level of non education completed by the contact person.	
	EmploymentType	O	O	Code that defines employment type. This is compulsory for Parents or Guardians of the Student.	

3.3.4.1.2 StudentContactRelationship

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	StudentPersonalRefId	M	M	GUID that identifies the student for the relationship.	
@	StudentContactPersonalRefId	M	M	GUID that identifies the contact person for the relationship.	
	StudentContactRelationshipRefId	O	M	GUID of the relationship record	

	Relationship	M	M	Defines the relationship of the contact to the student.	
	ContactFlags/ParentLegalGuardian	C	C	Does the individual have parental or legal guardianship responsibility for the student?	
	ContactFlags/PickupRights	C	C	This element tells whether or not the contact has pickup rights.	
	ContactFlags/LivesWith	C	C	Does the student live with this contact?	
	ContactFlags/AccessToRecords	C	C	Does this contact have access to the student's records?	
	ContactFlags/EmergencyContact	C	C	Should this contact be notified in case of emergency?	
	ContactFlags/HasCustody	C	C	Does this contact have or share custody of the student?	
	ContactFlags/DisciplinaryContact	C	C	Is this person to be contacted in case of disciplinary action?	
	ContactFlags/PrimaryCareProvider	C	C	Does this person provide daily living or personal assistance to the student?	
	ContactFlags/FeesBilling	C	C	Is this the contact responsible for Payment of Fees?	
	ContactFlags/FamilyMail	C	C	Does this person receive family mail?	
	ContactFlags/InterventionOrder	C	C	Is there an Intervention Order against this Contact?	

3.3.4.2 New Relationship added to a Current Contact

Summary: This scenario addresses the situation where a new relationship is added to an existing contact for example when a sibling starts at the school.

Pre-requisites: The StudentRefId and StudentContactPersonalRefId are already known.

Event	Description - Trigger	User Input	Agent Action - Publisher
New Contact Relationship	New contact relationship is added	School admin creates a new association between a current student and a current studentcontact.	Agent detects that a new relationship has been created and sends an ADD event for StudentContactRelationship.

Objects Events:

Object	Message	Behaviour	Choreography
StudentContactRelationship	Event	Create	Event to be published by SIS. It is recommended (but not required) that the event should be published in near real-time.

3.3.4.2.1 StudentContactRelationship

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	StudentPersonalRefId	M	M	GUID that identifies the student for the relationship.	
@	StudentContactPersonalRefId	M	M	GUID that identifies the contact person for the relationship.	
	StudentContactRelationshipRefId	O	M	GUID of the relationship record	
	Relationship	M	M	Defines the relationship of the contact to the student.	
	ContactFlags/ParentLegalGuardian	C	C	Does the individual have parental or legal guardianship responsibility for the student?	
	ContactFlags/PickupRights	C	C	This element tells whether or not the contact has pickup rights.	
	ContactFlags/LivesWith	C	C	Does the student live with this contact?	
	ContactFlags/AccessToRecords	C	C	Does this contact have access to the student's records?	
	ContactFlags/EmergencyContact	C	C	Should this contact be notified in case of emergency?	
	ContactFlags/HasCustody	C	C	Does this contact have or share custody of the student?	
	ContactFlags/DisciplinaryContact	C	C	Is this person to be contacted in case of disciplinary action?	
	ContactFlags/PrimaryCareProvider	C	C	Does this person provide daily living or personal assistance to the student?	
	ContactFlags/FeesBilling	C	C	Is this the contact responsible for Payment of Fees?	
	ContactFlags/FamilyMail	C	C	Does this person receive family mail?	
	ContactFlags/InterventionOrder	C	C	Is there an Intervention Order against this Contact?	

3.3.4.3 Change to a StudentContact Person information

Summary: This scenario addresses the situation where the information for an existing contact is changed.

Pre-requisites: None

Event	Description - Trigger	User Input	Agent Action - Publisher
Change to StudentContact person's information.	Information regarding an existing contact is changed.	School admin updates a contact's information.	Agent detects new contact and publishes a CHANGE event for StudentContactPersonal.

Objects Events:

Object	Message	Behaviour	Choreography
StudentContactPersonal	Event	Change	Event to be published by SIS. It is recommended (but not required) that the event should be published in near real-time.

3.3.4.3.1 StudentContactPersonal

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The ID (GUID) that uniquely identifies the instance of the object.	
	LocalId	O	O	The locally-assigned identifier for this person. Note: LocalId may be used as a unique person identifier rather than a unique contact identifier. In this case there may be more than one StudentContactPersonal instance with the same LocalId.	
	PersonInfo/Name/Title	O	O	A title or prefix associated with the name. If any of: Mr, Mrs, Ms, Miss, Rev, Fr, Dr, Prof, Hon, Sir, Lord, Lady - these must be as shown, otherwise free text. Note that title is not applicable to learners.	
	PersonInfo/Name/FamilyName	C	S	Family name. That part of the person's name which is used to describe family, clan, tribal group, or marital association. Note that this element is required when known. However, it may not be possible to know the family name in which case you should add the available information to the FullName element.	
	PersonInfo/Name/GivenName	C	S	Given name of the person. Note that this element is required if known.	
	PersonInfo/Name/MiddleName	C	S	All other given or middle names, each separated with a single space character.	
	PersonInfo/Name/PreferredFamilyName	C	C	The family name preferred most by the person (as written).	
	PersonInfo/Name/PreferredGivenName	C	C	The given name preferred most by the person (as written).	
	PersonInfo/Demographics/Sex	O	O	'Sex' is the distinction 'male' and 'female', as reported by the person	

	PersonInfo/AddressList/Address/Type	M	M	Code that defines the location of the address. Note: A subset of specific valid values for each instance in a data object may be listed in that object.	
	PersonInfo/AddressList/Address/Role	M	M	A facet of Address	
	PersonInfo/AddressList/Address/Street/Line1	M	M	Address line 1.	
	PersonInfo/AddressList/Address/Street/Line2	O	O	Address line 2.	
	PersonInfo/AddressList/Address/City	M	M	The city part of the address.	
	PersonInfo/AddressList/Address/StateProvince	C	C	The state or province code. StateTerritory Code only applicable if an Australian Address. Conditional- If Australian Address this is mandatory.	
	PersonInfo/AddressList/Address/PostalCode	M	M	The ZIP/postal code.	
	PersonInfo/AddressList/Address/Country	O	O	The country code.	
	PersonInfo/PhoneNumberList/PhoneNumber/Type	M	M	Code that specifies what type of phone number this is. Note: A subset of valid values may be specified in data objects.	
	PersonInfo/PhoneNumberList/PhoneNumber/Number	M	M	Phone number. Free-form, but typical Australian formats include: (###) ####-####, ####-####, +##### ### ###	
	PersonInfo/EmailList	O	S		EmailList is Optional, SBPGroup is strongly suggesting that at least one email address be sent.
	PersonInfo/EmailList/Email/Type	M	M	This attribute specifies the type of e-mail address	
	PersonInfo/EmailList/Email	M	M	This element represents an e-mail address of one of a number of types and occurs in objects such as StudentPersonal, StaffPersonal, StudentContactPersonal, etc.	

3.3.4.4 Change to StudentContact Relationship Info

Summary: This scenario addresses the situation where the relationship information for a Student's Contact has changed for example, when access rights have changed or living arrangements change.

Pre-requisites: None.

Event	Description - Trigger	User Input	Agent Action - Publisher
Change to StudentContactRelationship	Change in a student contact's relationship information.	School admin updates information associated with a student contact's relationship.	Agent detects a change in information that addresses the contact relationship between a contact and a student. E.g. change in pick-up rights.

Objects Events:

Object	Message	Behaviour	Choreography
StudentContactRelationship	Event	Change	Event to be published by SIS. It is recommended (but not required) that the event should be published in near real-time.

3.3.4.4.1 StudentContactRelationship

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	StudentPersonalRefId	M	M	GUID that identifies the student for the relationship.	
@	StudentContactPersonalRefId	M	M	GUID that identifies the contact person for the relationship.	
	StudentContactRelationshipRefId	O	M	GUID of the relationship record	
	Relationship	M	M	Defines the relationship of the contact to the student.	
	ContactFlags/ParentLegalGuardian	C	C	Does the individual have parental or legal guardianship responsibility for the student?	
	ContactFlags/PickupRights	C	C	This element tells whether or not the contact has pickup rights.	
	ContactFlags/LivesWith	C	C	Does the student live with this contact?	
	ContactFlags/AccessToRecords	C	C	Does this contact have access to the student's records?	
	ContactFlags/EmergencyContact	C	C	Should this contact be notified in case of emergency?	
	ContactFlags/HasCustody	C	C	Does this contact have or share custody of the student?	
	ContactFlags/DisciplinaryContact	C	C	Is this person to be contacted in case of disciplinary action?	
	ContactFlags/PrimaryCareProvider	C	C	Does this person provide daily living or personal assistance to the student?	

	ContactFlags/FeesBilling	C	C	Is this the contact responsible for Payment of Fees?	
	ContactFlags/FamilyMail	C	C	Does this person receive family mail?	
	ContactFlags/InterventionOrder	C	C	Is there an Intervention Order against this Contact?	

3.3.4.5 Remove a StudentContact from a Student

Summary: This scenario addresses the situation where the StudentContact person is removed from the student.

Pre-requisites: None.

Event	Description - Trigger	User Input	Agent Action - Publisher
Remove a StudentContact from a Student	Student	School Admin removes a StudentContact's relationship from a Student.	Agent detects change and sends a DELETE of the StudentContactRelationship Object. This only removes the relationship and not the actual Contact or Student.

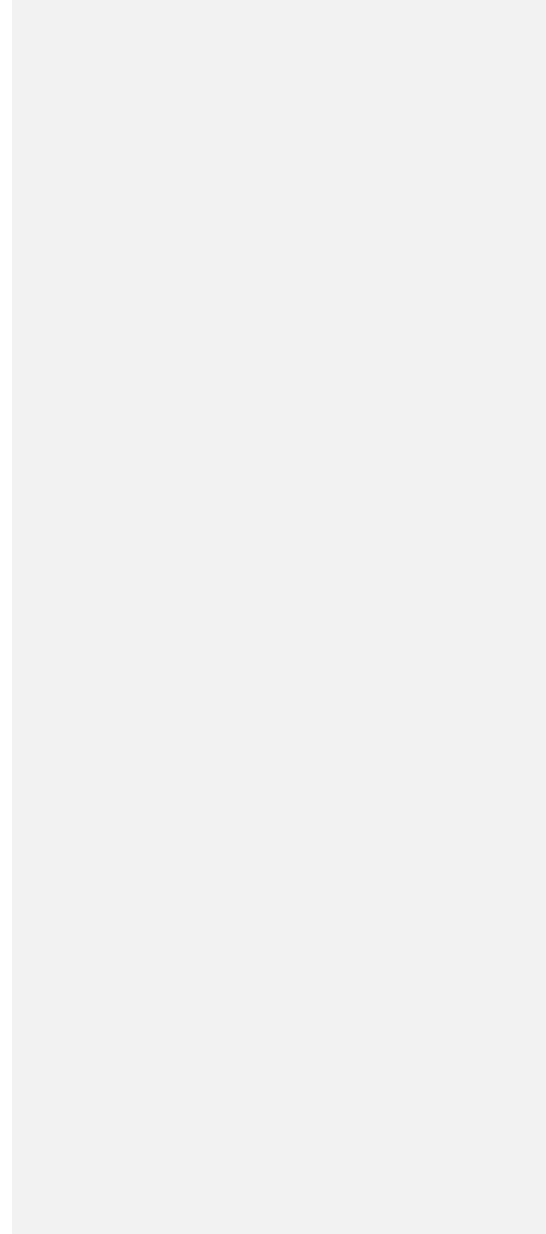
Objects Events:

Object	Message	Behaviour	Choreography
StudentContactRelationship	Event	Delete	Event to be published by SIS. It is recommended (but not required) that the event should be published in near real-time.

3.3.4.5.1 StudentContactRelationship

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	StudentPersonalRefId	M	M	GUID that identifies the student for the relationship.	
@	StudentContactPersonalRefId	M	M	GUID that identifies the contact person for the relationship.	
	StudentContactRelationshipRefId	O	M	GUID of the relationship record	

DRAFT until approved by DSWG



3.4. Class scenarios

3.4.0. Scenario

The SIS system is provisioning students and teacher attributes, and it is gathering information on what classes they are attending, as membership lists.

The SIS should be able to extract information in order to generate a period by period timetable for any given student or teacher, and report this information without significant dependence on the timetabling system (including re-exposing timetabling system objects which are not directly relevant to this goal).

This scenario does not need to deal with enrolments in multiple schools: the SIS is presumed specific to a single school, and only a student's or staff's assignments within that school are queried.

3.4.0.1 Scenarios out of scope

- a) A student or staff member could be provided with a period-by-period timetable of their assignments across all schools known of in the SIF Zone, and not just in a given school. This scenario is not addressed in the baseline.
- b) Where the TimeTable has been previously published via SIF in the zone, for any given timetabled class (TimeTableCell), the SIS should be able to retrieve from the timetabling system, and display, which students and which teachers are attending it. This can be done for staff by looking up respectively TimeTableCell/StaffPersonalRefId for staff; for students, it can be done by looking up TimeTableCell/TeachingGroupRefId, then all TeachingGroup/StudentList/TeachingGroupStudent/StudentLocalId for which TeachingGroup@RefId = TimeTableCell/TeachingGroupRefId.

These scenarios are not addressed further in the baseline.

3.4.1. Class-Centric information for publishing by an SIS

The TimeTabling Objects within the SIF Implementation Specification (Australia) 1.3 were designed to be published by a TimeTabling Package and consumed by an SIS; the scenario for using TeachingGroup assumes that the timetable is already in place on the zone, though it does not presuppose that it has been set up by a timetabling system in particular. As of version 1.2 of SIF AU, the specification of TeachingGroup includes a skeleton of what would be provided by the TimeTabling package in TimeTableCell and TimeTableSubject, in the element TeachingGroupPeriodList. As a result, the TeachingGroup object on its own can be published from an SIS as a provider to a Learning Management System, with enough information about a Class and its characteristics to be meaningful to the consuming system.

The scenario only reports TeachingGroup objects in response to queries; creating and updating TeachingGroup objects is discussed separately.

Objects Events:

Object	Message	Behaviour	Choreography
TeachingGroup	Event	Respond	
TeachingGroup	Event	Respond	

Four queries are covered for the object; two are mandatory (queries for a single-school SIF zone), two are strongly recommended (queries for a SIF zone covering multiple schools):

To generate a class membership list for a staff member within a school, where the SIF Zone only covers one school:

- Query by RefId all Teaching Groups for which the staff is included under TeachingGroup/TeacherList/TeachingGroupTeacher/StaffLocalId
- For each of these teaching groups,
 - Retrieve all associated TeachingGroup/TeachingGroupPeriodList instances
- (optional) Filter out all TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod instances for which the staff is **not** included under TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod/StaffLocalID

This results in one membership list per TeachingGroup that the staff member teaches, rather than a single membership list per staff member. By default the membership list includes all periods that each teaching group meets; the optional step limits the periods to those that the staff member is actually teaching, if the groups are taught by a team. The TeachingGroup/TeachingGroupPeriodList instances would need to be consolidated into a report, either through a new TimeTable object, or outside of SIF.

To generate a class membership list for a student within a school, where the SIF Zone only covers one school:

- Query by RefId all Teaching Groups for which the student is included under TeachingGroup/StudentList/TeachingGroupStudent/StudentLocalId
- For each of these teaching groups,
 - Retrieve all associated TeachingGroup/TeachingGroupPeriodList instances

This results in one schedule (TeachingGroup/TeachingGroupPeriodList) per TeachingGroup that the student is enrolled in, rather than a single membership list per student. The TeachingGroup/TeachingGroupPeriodList instances would need to be consolidated into a report, either through a new TimeTable object, or outside of SIF.

(Strongly recommended) To generate a class membership list for a staff member in a school, where the SIF Zone covers multiple schools:

- Query by RefId all Teaching Groups for which the staff is included under TeachingGroup/TeacherList/TeachingGroupTeacher/StaffPersonalRefId, and for which the school RefId matches TeachingGroup/SchoolInfoRefId
- For each of these teaching groups,
 - Retrieve all associated TeachingGroup/TeachingGroupPeriodList instances
- (optional) Filter out all TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod instances for which the staff is **not** included under TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod/StaffPersonalRefId

(Strongly recommended) To generate a class membership list for a student within a school, where the SIF Zone covers multiple schools:

- Query by RefId all Teaching Groups for which the student is included under TeachingGroup/StudentList/TeachingGroupStudent/StudentPersonalRefId, and for which the school RefId matches TeachingGroup/SchoolInfoRefId
- For each of these teaching groups,
 - Retrieve all associated TeachingGroup/TeachingGroupPeriodList instances

Assumptions:

- The SIS knows the GUIDs of students and teachers (as well as the Local IDs of staff), and does not need to retrieve StudentPersonal and StaffPersonal records externally.
- The SIS knows the GUIDs of the schools it covers, and does not need to retrieve SchoolInfo records externally.
- This scenario does not cover attendance for particular timetabled sessions (SessionInfo).
- TeachingGroup can record more staff as associated with the group on aggregated than are associated with any one period. (Hence the need to filter on TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod/StaffLocalID)
- Only one staff member is assigned to a given timetabled session.
 - That is all that TimeTableCell records. If multiple staff are assigned to a class, TeachingGroup will name the other staff members, but will not assign them to particular timetable cells.

Constraints:

Per the definition of TeachingGroup:

"All local ids in non-authoritative objects are optional. It is therefore up to the provider and the actual agent's design and choreography to determine whether or not to use or local ids from parent objects. There are cases in some subscribing systems where it might not possible to add RefId columns and therefore the RefId of related objects cannot be stored. For example a target system that listens to TimeTableSubject events and updates them may not be able to store the associated SchoolCourseInfoRefId with that object. The only way it can link the TimeTableSubject object with the appropriate course might be through its local course id. Having local ids that link the parent objects with the child object can simplify the agent design. Generally it is suggested to use the appropriate RefIds whenever possible and only use local ids if there is no other way to use RefIds."

As a result, both LocalIds and RefIds are provided, and RefIds are made mandatory.

3.4.1.1 TeachingGroup

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@K	RefId	M	M	The GUID of the TeachingGroup	

	SchoolYear	M	M	School year for which the information is applicable, expressed as the four-digit year in which the school year ends (e.g., "2007").	
	LocalId	M	M	LocalId of the Teaching Group (previously ClassIndicator).	
	ShortName	M	M	Short free format label that describes the group.	
	SchoolInfoRefId	O	S	GUID of SchoolInfo object this teaching group belongs to.	
	SchoolLocalId	O	O	Local School Id.	
	TimeTableSubjectRefId	O	O	GUID of TimeTableSubject object this teaching group belongs to	
	TimeTableSubjectLocalId	O	O	Local subject Id.	
	StudentList	O	M	List of Students in this Teaching Group	
	StudentList/TeachingGroupStudent	MR	MR	Students who belong to the TeachingGroup	
	StudentList/TeachingGroupStudent/StudentPersonalRefId	O	O	GUID from the StudentPersonal or StudentSnapshot Object (if known), that identifies the student.	
	StudentList/TeachingGroupStudent/StudentLocalId	M	M	Local Id of the Student	
	StudentList/TeachingGroupStudent/Name	M	M	Name of student	
	TeacherList	O	S	A List of teachers associated with the group – may or may not include the teacher timetabled in the schedule.	
	TeacherList/TeachingGroupTeacher	MR	MR	Teacher associated with the Teaching group – may or may not be the teacher timetabled in the schedule	
	TeacherList/TeachingGroupTeacher/StaffPersonalRefId	O	M	GUID from the StaffPersonal Object that identifies the Staff member	
	TeacherList/TeachingGroupTeacher/StaffLocalId	M	M	LocalId of the Staff member	
	TeacherList/TeachingGroupTeacher/Name	M	M		
	TeacherList/TeachingGroupTeacher/Association	M	M	An association with the Teaching Group.	
	TeachingGroupPeriodList	O	S	A List of scheduled meeting times for this Teaching Group	
	TeachingGroupPeriodList/TeachingGroupPeriod	O	S	A scheduled meeting time for this Teaching	

				Group	
	TeachingGroupPeriodList/TeachingGroupPeriod/TimeTableCellRefId	O	O	If the TimeTableCell has been previously published, this reference can be provided as a connection to more detailed schedule information.	
	TeachingGroupPeriodList/TeachingGroupPeriod/RoomNumber	O	O	Room number as presented to the user/application.	
	TeachingGroupPeriodList/TeachingGroupPeriod/StaffLocalId	O	O	Staff LocalId	
	TeachingGroupPeriodList/TeachingGroupPeriod/DayId	M	M	Day Id	
	TeachingGroupPeriodList/TeachingGroupPeriod/PeriodId	C	C	Period Id within the Day Id. Either Period Id or starting time must be provided.	
	TeachingGroupPeriodList/TeachingGroupPeriod/StartTime	C	C	Starting time of class within the Day Id. Either Period Id or starting time must be provided.	
	TeachingGroupPeriodList/TeachingGroupPeriod/CellType	O	O	Type of Lesson/Session e.g. Teaching, Lunch etc.	

3.4.2. Provisioning Teaching Group periods

The scenario in 3.4.1. is driven by TeachingGroup/TeachingGroupPeriodList to communicate a staff or student's scheduled commitments in a school. Any change to staff or student's association within teaching groups will result in a change of their reported scheduling. Any change to other elements of TeachingGroup (such as the short name) will also by definition, result in a change event for TeachingGroup.

The purpose of the period list is to provide consuming systems such as portals and VLE/ LMS or online applications that wish to group teachers and students by classes and period groups. It is NOT intended to provision a timetabling system or to be used to provision a SIS from a timetabling system.

It is assumed that TeachingGroup/TeachingGroupPeriodList is pre-determined by a timetabling system, and TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod can be dependent on that system. It is also assumed that it can, in a full schedule or TimeTable provisioning be represented as part of TimeTableCell (indicated through TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod/TimeTableCellRefId).

Any updates to those TimeTableCell instances will also need to be reflected in the dependent TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod. Note that any changes to TeachingGroupPeriodList/TeachingGroupPeriod do **not** trigger a change on the corresponding TimeTableCell entity: any referenced TimeTableCell entity is the source of truth for student scheduling, and updates must only be in one direction.

This set of scenarios requires notification of enrolment changes for teachers and students in teaching groups, so that it can update its membership lists. The creation of a TeachingGroup instance by itself does not impact staff or student scheduling until TeachingGroupPeriodList and TeacherList or StudentList are provisioned.

Objects Events:

Object	Message	Behaviour	Choreography
TimeTableCell	Event	Change	update TeachingGroup/TeachingGroupPeriodList
TimeTableCell	Event	Create	update TeachingGroup/TeachingGroupPeriodList
TeachingGroup	Event	Change	
TeachingGroup	Event	Create	

3.4.3. Teaching Group set up

Summary: A teaching group is created, and contains enough information that it can be used to report on scheduling (StudentList or TeacherList, and TeachingGroupPeriodList, each with at least one populated entry).

For setting the teaching group up, there is no requirement for a timetabling system to be used to provision the TimeTableCell data: the data could for example be loaded in from a CSV file.

Pre-requisites:

- Either TeachingGroup/StudentList or TeachingGroup/TeacherList contain at least one valid StudentLocalId or StaffLocalId instance
- TeachingGroup/TeachingGroupPeriodList contains at least one non-empty instance of TeachingGroupPeriod. Note that this is a requirement specific to the timetabling scenarios discussed here; it is not a requirement for TeachingGroup in SIF AU in general.
- The SIS already has been given access to class information – although this need not have been provisioned directly from a timetabling system.

Objects Events:

Object	Message	Behaviour	Choreography
TeachingGroup	Event	Create	

3.4.4. Staff Joins Teaching Group

Summary: A staff member joins a teaching group. They are added to TeachingGroup/TeacherList, which makes the teaching group subject to scheduling queries for that staff member. The staff member may also be added to individual TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod/StaffLocalId instances at this stage. These TeachingGroupPeriod instances may be managed as instances of TimeTableCell, by timetabling software.

Pre-requisites:

- Teaching group object already exists.
- Staff already has a local ID

Objects Events:

Object	Message	Behaviour	Choreography
TimeTableCell	Event	Change	If TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod/ references staff member, and TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod/ references TimeTableCellRefId, then any relevant updates to the source TimeTableCell entity should be published before the referencing TeachingGroup instance

Note: Where the Teacher has been referenced in a previously published Object (TimeTableCell - not in the baseline), there is no requirement to re-publish TimeTableCell in the context of the SBP. For this profile the SIS is the provider of this information - even if the initial source was a TimeTabling package.

Note Also: That if any member of the TeachingGroupTeacherList changes - that the Object needs to be re-published with a new TeacherList containing all the current TeachingGroupTeachers.

3.4.4.1 Teaching Group

Attribute/ Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@K	RefId	M	M	The GUID of the TeachingGroup	
	SchoolYear	M	M	School year for which the information is applicable, expressed as the four-digit year in which the school year ends (e.g., "2007").	
	LocalId	M	M	LocalId of the Teaching Group (previously ClassIndicator).	
	ShortName	M	M	Short free format label that describes the group.	
	SchoolInfoRefId	O	S	GUID of SchoolInfo object this teaching group belongs to.	
	TimeTableSubjectRefId	O	O	GUID of TimeTableSubject object this teaching group belongs to	

	StudentList	O	M	List of Students in this Teaching Group	
	TeacherList	O	S	A List of teachers associated with the group – may or may not include the teacher timetabled in the schedule.	
	TeacherList/TeachingGroupTeacher	MR	MR	Teacher associated with the Teaching group – may or may not be the teacher timetabled in the schedule	
	TeacherList/TeachingGroupTeacher/StaffPersonalRefId	O	M	GUID from the StaffPersonal Object that identifies the Staff member	
	TeacherList/TeachingGroupTeacher/StaffLocalId	M	M	LocalId of the Staff member	
	TeacherList/TeachingGroupTeacher/Name	M	M		
	TeacherList/TeachingGroupTeacher/Association	M	M	An association with the Teaching Group.	
	TeachingGroupPeriodList	O	S	A List of scheduled meeting times for this Teaching Group	
	TeachingGroupPeriodList/TeachingGroupPeriod	O	S	A scheduled meeting time for this Teaching Group	
	TeachingGroupPeriodList/TeachingGroupPeriod/TimeTableCellRefId	O	O	If the TimeTableCell has been previously published, this reference can be provided as a connection to more detailed schedule information.	
	TeachingGroupPeriodList/TeachingGroupPeriod/RoomNumber	O	O	Room number as presented to the user/application.	
	TeachingGroupPeriodList/TeachingGroupPeriod/StaffLocalId	O	O	Staff LocalId	
	TeachingGroupPeriodList/TeachingGroupPeriod/DayId	M	M	Day ID	
	TeachingGroupPeriodList/TeachingGroupPeriod/PeriodId	C	C	Period Id within the Day Id. Either Period Id or starting time must be provided.	
	TeachingGroupPeriodList/TeachingGroupPeriod/StartTime	C	C	Starting time of class within the Day Id. Either Period Id or starting time must be provided.	
	TeachingGroupPeriodList/TeachingGroupPeriod/CellType	O	O	Type of Lesson/Session eg Teaching, Lunch etc	

3.4.5. Staff Leaves Teaching Group

Summary: A staff member leaves a teaching group, whether because they are assigned to other commitments, or because they leave the school. The staff member is removed from any remaining individual TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod/StaffLocalId instances. (This may involve changing source TimeTableCell instances.) The staff member is removed from TeachingGroup/TeacherList.

Pre-requisites:

- Teaching group object already exists.
- Staff member is already referenced in TeachingGroup/TeacherList

Note Also: If any member of the TeachingGroupTeacherList changes, the Object needs to be re-published with a new TeacherList containing all the current TeachingGroupTeachers.

Objects Events:

Object	Message	Behaviour	Choreography
TeachingGroup	Event	Change	Event should be published after references to the staff member are removed from all TeachingGroupPeriod instances within the object, as well as from TeacherList

3.4.5.1 Teaching Group

Attribute/ Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@K	RefId	M	M	The GUID of the TeachingGroup	
	SchoolYear	M	M	School year for which the information is applicable, expressed as the four-digit year in which the school year ends (e.g., "2007").	
	LocalId	M	M	LocalId of the Teaching Group (previously ClassIndicator).	
	ShortName	M	M	Short free format label that describes the group.	
	SchoolInfoRefId	O	S	GUID of SchoolInfo object this teaching group belongs to.	
	TimeTableSubjectRefId	O	O	GUID of TimeTableSubject object this teaching group belongs to	
	StudentList	O	M	List of Students in this Teaching Group	
	TeacherList	O	S	A List of teachers associated with the group – may or may not include the teacher timetabled in the schedule.	
	TeacherList/TeachingGroupTeacher	MR	MR	Teacher associated with the Teaching group – may or may not be the teacher timetabled in the schedule	
	TeacherList/TeachingGroupTeacher/StaffPersonalRefId	O	M	GUID from the StaffPersonal Object that identifies the Staff member	
	TeacherList/TeachingGroupTeacher/StaffLocalId	M	M	LocalId of the Staff member	
	TeacherList/TeachingGroupTeacher/Name	M	M		
	TeacherList/TeachingGroupTeacher/Association	M	M	An association with the Teaching Group.	

	TeachingGroupPeriodList	O	S	A List of scheduled meeting times for this Teaching Group	
	TeachingGroupPeriodList/TeachingGroupPeriod	O	S	A scheduled meeting time for this Teaching Group	
	TeachingGroupPeriodList/TeachingGroupPeriod/TimeTableCellRefId	O	O	If the TimeTableCell has been previously published, this reference can be provided as a connection to more detailed schedule information.	
	TeachingGroupPeriodList/TeachingGroupPeriod/RoomNumber	O	O	Room number as presented to the user/application.	
	TeachingGroupPeriodList/TeachingGroupPeriod/StaffLocalId	O	O	Staff LocalId	
	TeachingGroupPeriodList/TeachingGroupPeriod/DayId	M	M	Day ID	
	TeachingGroupPeriodList/TeachingGroupPeriod/PeriodId	C	C	Period Id within the Day Id. Either Period Id or starting time must be provided.	
	TeachingGroupPeriodList/TeachingGroupPeriod/StartTime	C	C	Starting time of class within the Day Id. Either Period Id or starting time must be provided.	
	TeachingGroupPeriodList/TeachingGroupPeriod/CellType	O	O	Type of Lesson/Session eg Teaching, Lunch etc	

3.4.6. Staff Assignments Change in Teaching Group

Summary: A staff member changes period allocation within a teaching group (or else, is given period allocations for the first time). The staff member assignment to particular TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod/StaffLocalId instances is changed. (This may involve changing source TimeTableCell instances.) The description of the staff member's role in TeacherList/TeachingGroupTeacher/Association may also change.

Pre-requisites:

Teaching group object already exists.

Teaching group period list already exists, or else underlying TimeTableCell instances already exists

Staff member is already referenced in TeachingGroup/TeacherList

Objects Events:

Object	Message	Behaviour	Choreography
TeachingGroup	Event	Change	

Note Also: If any member of the TeachingGroupTeacherList changes, the Object needs to be re-published with a new TeacherList containing all the current TeachingGroupTeachers.

3.4.6.1 Teaching Group

Attribute/ Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@K	RefId	M	M	The GUID of the TeachingGroup	
	SchoolYear	M	M	School year for which the information is applicable, expressed as the four-digit year in which the school year ends (e.g., "2007").	
	LocalId	M	M	LocalId of the Teaching Group (previously ClassIndicator).	
	ShortName	M	M	Short free format label that describes the group.	
	SchoolInfoRefId	O	S	GUID of SchoolInfo object this teaching group belongs to.	
	TimeTableSubjectRefId	O	O	GUID of TimeTableSubject object this teaching group belongs to	
	StudentList	O	M	List of Students in this Teaching Group	
	TeacherList	O	S	A List of teachers associated with the group – may or may not include the teacher timetabled in the schedule.	
	TeacherList/TeachingGroupTeacher	MR	MR	Teacher associated with the Teaching group – may or may not be the teacher timetabled in the schedule	
	TeacherList/TeachingGroupTeacher/StaffPersonalRefId	O	M	GUID from the StaffPersonal Object that identifies the Staff member	
	TeacherList/TeachingGroupTeacher/StaffLocalId	M	M	LocalId of the Staff member	
	TeacherList/TeachingGroupTeacher/Name	M	M		
	TeacherList/TeachingGroupTeacher/Association	M	M	An association with the Teaching Group.	
	TeachingGroupPeriodList	O	S	A List of scheduled meeting times for this Teaching Group	
	TeachingGroupPeriodList/TeachingGroupPeriod	O	S	A scheduled meeting time for this Teaching Group	
	TeachingGroupPeriodList/TeachingGroupPeriod/TimeTableCellRefId	O	O	If the TimeTableCell has been previously published, this reference can be provided as a connection to more detailed schedule information.	
	TeachingGroupPeriodList/TeachingGroupPeriod/RoomNumber	O	O	Room number as presented to the user/application.	
	TeachingGroupPeriodList/TeachingGroupPeriod/StaffLocalId	O	O	Staff LocalId	
	TeachingGroupPeriodList/TeachingGroupPeriod/DayId	M	M	Day ID	
	TeachingGroupPeriodList/TeachingGroupPeriod/PeriodId	C	C	Period Id within the Day Id. Either Period Id or starting time must be provided.	

	TeachingGroupPeriodList/TeachingGroupPeriod/StartTime	C	C	Starting time of class within the Day Id. Either Period Id or starting time must be provided.	
	TeachingGroupPeriodList/TeachingGroupPeriod/CellType	O	O	Type of Lesson/Session eg Teaching, Lunch etc	

3.4.7. Student Joins Teaching Group

Summary: A student joins a teaching group. They are added to TeachingGroup/StudentList, which makes the teaching group subject to scheduling queries for that student.

Pre-requisites:

Teaching group object already exists.

Student already has a local ID

Objects Events:

Object	Message	Behaviour	Choreography
TeachingGroup	Event	Change	

Note Also: If any member of the TeachingGroup StudentList changes, the Object needs to be re-published with a new StudentList containing all the current TeachingGroup Students..

3.4.7.1 TeachingGroup

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@K	RefId	M	M	The GUID of the TeachingGroup	
	SchoolYear	M	M	School year for which the information is applicable, expressed as the four-digit year in which the school year ends (e.g., "2007").	
	LocalId	M	M	LocalId of the Teaching Group (previously ClassIndicator).	
	ShortName	M	M	Short free format label that describes the group.	
	TimeTableSubjectRefId	O	O	GUID of TimeTableSubject object this teaching group belongs to	
	StudentList	O	M	List of Students in this Teaching Group	
	StudentList/TeachingGroupStudent	MR	MR	Students who belong to the TeachingGroup	

	StudentList/TeachingGroupStudent/StudentPersonalRefId	O	M	GUID from the StudentPersonal Object (if known), that identifies the student.
	StudentList/TeachingGroupStudent/StudentLocalId	M	M	Local Id of the Student
	StudentList/TeachingGroupStudent/Name	M	M	Name of student
	TeacherList	O	S	A List of teachers associated with the group – may or may not include the teacher timetabled in the schedule.
	TeachingGroupPeriodList	O	S	A List of scheduled meeting times for this Teaching Group
	TeachingGroupPeriodList/TeachingGroupPeriod	O	S	A scheduled meeting time for this Teaching Group
	TeachingGroupPeriodList/TeachingGroupPeriod/TimeTableCellRefId	O	O	If the TimeTableCell has been previously published, this reference can be provided as a connection to more detailed schedule information.
	TeachingGroupPeriodList/TeachingGroupPeriod/RoomNumber	O	O	Room number as presented to the user/application.
	TeachingGroupPeriodList/TeachingGroupPeriod/StaffLocalId	O	O	Staff LocalId
	TeachingGroupPeriodList/TeachingGroupPeriod/DayId	M	M	Day ID
	TeachingGroupPeriodList/TeachingGroupPeriod/PeriodId	C	C	Period Id within the Day Id. Either Period Id or starting time must be provided.
	TeachingGroupPeriodList/TeachingGroupPeriod/StartTime	C	C	Starting time of class within the Day Id. Either Period Id or starting time must be provided.
	TeachingGroupPeriodList/TeachingGroupPeriod/CellType	O	O	Type of Lesson/Session eg Teaching, Lunch etc

3.4.8. Student Leaves Teaching Group

Summary: A student leaves a teaching group. They are removed from TeachingGroup/StudentList.

Pre-requisites:

- Teaching group object already exists.
- Student is referenced from TeachingGroup/StudentList

Objects Events:

Object	Message	Behaviour	Choreography
--------	---------	-----------	--------------

TeachingGroup	Event	Change	
---------------	-------	--------	--

Note Also: If any member of the TeachingGroup StudentList changes, the Object needs to be re-published with a new StudentList containing all the current TeachingGroup Students.

Attribute/Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@K	RefId	M	M	The GUID of the TeachingGroup	
	SchoolYear	M	M	School year for which the information is applicable, expressed as the four-digit year in which the school year ends (e.g., "2007").	
	LocalId	M	M	LocalId of the Teaching Group (previously ClassIndicator).	
	ShortName	M	M	Short free format label that describes the group.	
	TimeTableSubjectRefId	O	O	GUID of TimeTableSubject object this teaching group belongs to	
	StudentList	O	M	List of Students in this Teaching Group	
	StudentList/TeachingGroupStudent	MR	MR	Students who belong to the TeachingGroup	
	StudentList/TeachingGroupStudent/StudentPersonalRefId	O	M	GUID from the StudentPersonal Object (if known), that identifies the student.	
	StudentList/TeachingGroupStudent/StudentLocalId	M	M	Local Id of the Student	
	StudentList/TeachingGroupStudent/Name	M	M	Name of student	
	TeacherList	O	S	A List of teachers associated with the group – may or may not include the teacher timetabled in the schedule.	

3.4.9. Period in Teaching Group Rescheduled

Summary: A period within a teaching group is added, is removed, or changes (including changing scheduled time or day, or changing description). TeachingGroup/TeachingGroupPeriodList/TeachingGroupPeriod is updated. (This may involve changing source TimeTableCell instances.)

Pre-requisites:

- Teaching group object already exists.
- Teaching group period list already exists, or else underlying TimeTableCell instances already exists

Note Also: That if any member of the TeachingGroupTeacherList or PeriodList changes - that the Object needs to be re-published with a new TeacherList containing all the current TeachingGroupTeachers and all the current PeriodList periods.

Objects Events:

Object	Message	Behaviour	Choreography
TeachingGroup	Event	Change	Note that any changes to TeachingGroupPeriodList/TeachingGroupPeriod do not trigger a change on the corresponding TimeTableCell entity: any referenced TimeTableCell entity is the source of truth for student scheduling, and updates must only be in one direction.

3.4.9.1 Teaching Group

Attribute/ Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@K	RefId	M	M	The GUID of the TeachingGroup	
	SchoolYear	M	M	School year for which the information is applicable, expressed as the four-digit year in which the school year ends (e.g., "2007").	
	LocalId	M	M	LocalId of the Teaching Group (previously ClassIndicator).	
	ShortName	M	M	Short free format label that describes the group.	
	SchoolInfoRefId	O	S	GUID of SchoolInfo object this teaching group belongs to.	
	TimeTableSubjectRefId	O	O	GUID of TimeTableSubject object this teaching group belongs to	
	StudentList	O	M	List of Students in this Teaching Group	
	TeacherList	O	S	A List of teachers associated with the group – may or may not include the teacher timetabled in the schedule.	
	TeacherList/TeachingGroupTeacher	MR	MR	Teacher associated with the Teaching group – may or may not be the teacher timetabled in the schedule	
	TeacherList/TeachingGroupTeacher/StaffPersonalRefId	O	M	GUID from the StaffPersonal Object that identifies the	

				Staff member	
	TeacherList/TeachingGroupTeacher/StaffLocalId	M	M	LocalId of the Staff member	
	TeacherList/TeachingGroupTeacher/Name	M	M		
	TeacherList/TeachingGroupTeacher/Association	M	M	An association with the Teaching Group.	
	TeachingGroupPeriodList	O	S	A List of scheduled meeting times for this Teaching Group	
	TeachingGroupPeriodList/TeachingGroupPeriod	O	S	A scheduled meeting time for this Teaching Group	
	TeachingGroupPeriodList/TeachingGroupPeriod/TimeTableCellRefId	O	O	If the TimeTableCell has been previously published, this reference can be provided as a connection to more detailed schedule information.	
	TeachingGroupPeriodList/TeachingGroupPeriod/RoomNumber	O	O	Room number as presented to the user/application.	
	TeachingGroupPeriodList/TeachingGroupPeriod/StaffLocalId	O	O	Staff LocalId	
	TeachingGroupPeriodList/TeachingGroupPeriod/DayId	M	M	Day ID	
	TeachingGroupPeriodList/TeachingGroupPeriod/PeriodId	C	C	Period Id within the Day Id. Either Period Id or starting time must be provided.	
	TeachingGroupPeriodList/TeachingGroupPeriod/StartTime	C	C	Starting time of class within the Day Id. Either Period Id or starting time must be provided.	
	TeachingGroupPeriodList/TeachingGroupPeriod/CellType	O	O	Type of Lesson/Session eg Teaching, Lunch etc	

4. Optional Objects – may include

4.1. PersonPicture

This Object is Optional and is added for completeness when required.

This is the Picture of either the Student or the Staff member being published and should be published after the StudentPersonal or StaffPersonal Object has been published in the zone.

Attribute/ Key Identifier	Element/Attribute	Char	Baseline Char	Description	Business Rules
@	RefId	M	M	The ID (GUID) that uniquely identifies the picture.	

	ParentObjectRefId	M	M	The GUID of the personal object to which this picture is linked.	
@	ParentObjectRefId\SIF_RefObject	M	M	The name of the object reference.	Student with StudentSchoolEnrollment
	SchoolYear	M	M	School year for which this enrollment is applicable, expressed as the four-digit year in which the school year ends (e.g. 2007). StudentSchoolEnrollment instances must not span multiple school years.	
	PictureSource	M	M	This element defines the picture. If the Type attribute is URL, this is the location of the picture in [JPEG] format; if Type is JPEG, this is the [JPEG] image data encoded using the Base64 Content-Transfer-Encoding defined in Section 6.8 of [RFC 2045]	
@	PictureSouce\Type	M	M	The way the picture is specified.	jpg
	OKToPublish	O	M	Can the picture be published?	Need to know whether it is publishable – must be sent.

DRAFT until approved

