Table of Contents

================================== Query-by-example or Service Paths? ==================================

The following considerations apply in choosing whether to implement queries as Query By Example (QBE) or Service Paths (SP):

**Part of SIF standard and SIF Common framework**

QBE is not yet part of the SIF standard, but it will be included in the next release of the SIF Infrastructure specification, scheduled for January 2016, and is already included in the SIF Framework. Service paths are already part of the SIF standard.

**Service Paths are not predefined in SIF**

Service Paths are not predefined in the SIF specification: every instance of a SIF server needs to define the Service Paths it will support, through a SIF binding document, and advertise their availability through SIF Infrastructure services.

QBE is defined as a template capability, but is also not supported by default in SIF servers: a QBE mapping needs to be realised for each object.

**Service Paths best as a query joined on ID**

The semantics of Service Paths is potentially opaque, and needs to be defined in the SIF binding.

Best practice is to treat the Service Path X/{ID}/Y as a join query between X and Y, joined on ID, where ID is the single query parameter. If there are multiple possible RefIds on which the join can take place, the specific RefIDs need to be made explicit and agreed between users.

Best practice is not to include other query constraints in the query, explicitly or implicitly. This is to reduce guesswork from external users about what the Service Path query means. For example, it is not best practice to include in a Service Path /SchoolInfo/{ID}/StudentPersonal the constraint that school enrolments must be current, that school enrolments apply to the current school year, or that the schools are active. Naive users cannot anticipate these additional constraints on the query. While such added constraints can be added to Service Path queries, best practice is to use Query By Example instead.

Potential and valid exceptions are through the use of contexts where a "current", "past" or "future" context could be used to define temporally based information. For example the Service Path /SchoolInfo/{ID}/StudentPersonal in a "current" context could mean only currently enrolled students at the given school.

**Use Service Paths if implicit ID**

If a Service Path query X/{ID}/Y involves an ID which is not an explicit and mandatory element of Y, then Service Paths are the only way to realise the query: QBE is not applicable.

* For example, /SchoolInfo/{ID}/StudentPersonal depends on a join mediated by StudentSchoolEnrollment: neither StudentPersonal nor SchoolInfo refer directly to each other, so the query cannot be realised through QBE on a StudentPersonal or SchoolInfo object.
* For example, /StudentPersonal/{ID}/Timetable depends on a join via TimeTable/SchoolInfoRefId, which is an optional element, and may not be populated. However, TimeTableCell/RoomInfoRefId is mandatory, and RoomInfo/SchoolInfoRefId is mandatory, so the SchoolInfo for a non-empty TimeTable can always be recovered. For that reason, the query should be realised through SP rather than QBE.

**Use either if explicit ID...**

If a Service Path query X/{ID}/Y involves ID which is an explicit and mandatory element of Y, then it can be realised as either a Service Path, or as a QBE on Y against ID, and not involving X at all. For example, a query for StudentPersonal/{ID}/PersonPicture can be replaced by a QBE on PersonPicture/ParentObjectRefId = ID and PersonPicture/ParentObjectRefId@SIF\_RefObject = StudentPersonal.

**...but consider this...**

1. In case the query can be realised as either SP or QBE, the following considerations apply:

* Each Service Path needs to be implemented separately. The QBE for an object needs to be implemented only once. So if there are multiple Service Paths for an object X, QBE involves less resources to be managed.
* Each Service Path has its own ACL. The QBE for an object involves a single ACL. So if there needs to be differential authorisation for different elements n the object, the QBE needs additional resources to be implemented, to managed access to elements at a separate layer.

**HITS does QBE by default**

1. The HITS default where a query can be realised as either SP or QBE is to realise it as a QBE, as this involves less resources for the HITS team to manage. This may be overridden depending on implementation considerations for particular queries. SIF implementations will need to make their own decisions about whether to use SP or QBE.