Solution 2: A US Wide Methodology for Supporting Multiple Production VERSIONS of FHIR

As the FHIR standard usage becomes ubiquitous across the nation, the healthcare ecosystem will need to support multiple versions of FHIR resources and their associated APIs. Unlike the classic world of healthcare interoperability in which versions rarely changed (and when it did, everyone changed at once), modern API patterns support multiple versions at the same time. Regardless of what FHIR versions are included in regulation, or the version adoption rules that stem from ONC and CMS rules, the industry is going to wind up with multiple versions of FHIR and multiple versions of resources, extensions, profiles, value sets, and implementation guides.

In today's environment, most FHIR endpoints only support one version of FHIR, and there are multiple incompatible versions of FHIR in production (eg, DSTU2, STU3, R4). These versions are not fully backward or forward compatible, which means that incompatible or "breaking" changes may exist between versions (ie, changes in one FHIR version that would cause a system or application using a different FHIR version to fail) except where FHIR resources are "normative" (ie, content is considered stable and compatible between versions). While mapping solutions exist for some FHIR resources to convert from one version to another, their capability, quality, and completeness vary from resource to resource. In addition, there is no definitive source for transforms.

Initiators of transactions must be able to determine the version(s) available at the endpoint from which they are requesting information or know that the endpoint can handle multiple versions (or be version-agnostic) so that they can request and receive data effectively (ie, "speak the same language"). Some organizations may be able to maintain data in one format and dynamically convert to the necessary FHIR version on demand, but many organizations store data in the format (ie, version) in which it was received. In order for those organizations to make data available to other entities that require different versions of FHIR, they will likely have to manage multiple representations and corresponding mappings of the same information for different FHIR versions. This model does not scale well to support a large number of exchange partners and multiple versions of FHIR artifacts.

The *FAST* team has proposed a solution for how to handle version control more effectively to handle these scenarios, potential tooling to address them, and recommendations for what organizations can do to mitigate these issues.

In an ideal future state, relevant FHIR artifacts would be normative and any variation between FHIR releases would be focused on new functionality or edge cases. New FHIR versions would be backward compatible for all normative content, and all FHIR artifacts (eg, resources, profiles, bundles) would provide version information as part of any exchange. There would also need to be policies and tooling in place to support migration to new "floor" versions of FHIR as they evolve (ie, the minimum standard implementers must meet per regulation), such as a two-year window to sunset an old version, identification of any incompatible changes between new and old versions, and HL7 tools to translate between them.

In the interim, progress can be made toward these goals with improvements in resource version identification, capability statements (ie, documentation of the functionality supported for specific FHIR versions), and tooling such as authoritative mappings across versions. Since multiple versions of FHIR are currently in production, it will be important for organizations to be able to identify what version of FHIR their exchange partners are using so they may communicate using the same version or translate to the version supported by their exchange partner if needed. When an organization performs a directory lookup for an endpoint, all directory entries should include information regarding the FHIR version(s) supported. This requirement to support endpoint version is being accounted for in the *FAST* Endpoint Directory solution and will be incorporated into the HL7 Implementation Guide(s) being developed for the exchange of directory information. In addition, as the FHIR standard continues to evolve, organizations may support different functions at different times, making the capability statement an essential component to determine current endpoint support for specific versions and functionality. All endpoints will need to support the capability statement query and the FHIR $versions operation that returns the supported version(s). While the FHIR Capability Statement resource is normative, there are elements included within it that are not, and so it will also need to be updated to ensure that it does not change in significant ways from FHIR release to release.

As new versions of FHIR get released with incompatible changes between versions, HL7 tools will be needed to handle translation or mapping between old and new FHIR versions. Implementers will need these translation mappings tools to reconcile any differences as they communicate with exchange partners using different FHIR versions.

Solution 2: A US Wide Methodology for Supporting Multiple Production VERSIONS of FHIR

Quick Reference Summary of Barrier and Solution With Links to Resources

BARRIER

There are multiple incompatible versions of FHIR in production (eg, DSTU2, STU3, R4) with breaking changes between them. Until FHIR becomes "normative" (ie, content is stable between versions), the industry will need a way to manage different representations of the same information in different FHIR versions.

SOLUTION

- Resource version identification:
  - standard requirements for resources, profiles, and bundles
  - directory metadata for endpoints
  - capability statements
  - testing and validation for conformance to directory metadata and capability statements
- Support for multiple versions:
  - ability to identify endpoint version
  - ability to identify FHIR artifact version
  - ability to translate versions (at least from prior to new version)
  - Ability to document translations where appropriate (eg, provenance)

OPEN ITEMS

- Collaborate with HL7 FHIR leadership to ensure alignment with FHIR standard release plans and impact on extensions, profiles and Implementation Guides
- Identify impact on current and future ONC and CMS regulations
- Determine ability to translate non-normative resources
- Determine how version management works over time in response to new data portability requirements.
- Need to consider proliferation of Implementation Guides and Profiles that provide for different solutions to the same fundamental use case
- Consider incompatible profile constraints on underlying resource in ways that do not permit reuse by other implementation guides (eg, US Core constraints that do not support specific IGs requirements)

IN SCOPE

Managing multiple versions of FHIR and FHIR artifacts such as implementation guides, identification of supported version for a specific endpoint, transform/ translation service considerations, and the ability to appropriately manage exchange of information across multiple versions of FHIR.

OUT OF SCOPE

Specifying a single version of FHIR, requiring forward/ backward compatibility for non-normative resources, addressing support for multiple versions in a single exchange.

BENEFIT

- Clarity for implementers on version management methodology
- Efficiency through establishment of authoritative source for transforms/mappings across versions
- Improved reliability of FHIR message processing

STATUS

Finalizing V2 solution documentation to obtain SME input

CURRENT SOLUTION

Versioning Solution Document (Draft V2)

Solution 2: A US Wide Methodology for Supporting Multiple Production VERSIONS of FHIR
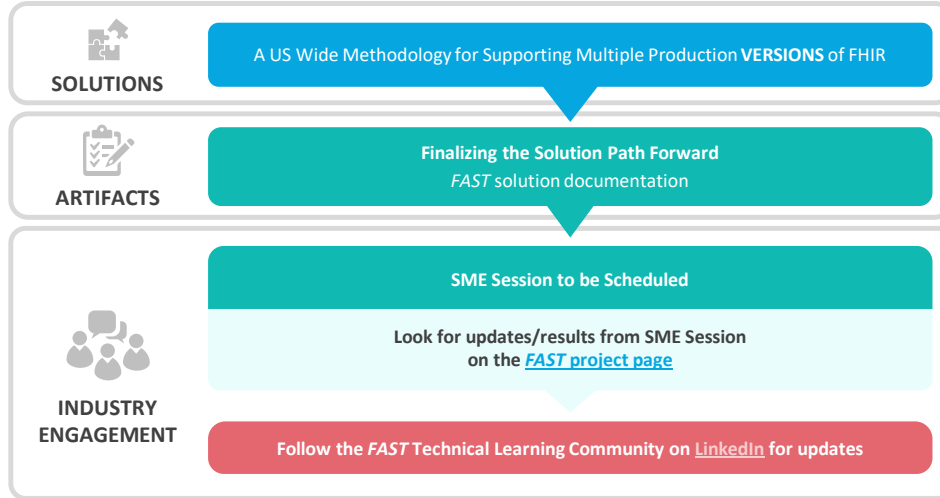
In the following diagram an entity, referred to as a requestor actor, needs to obtain information from another entity, known as the responder actor. The requestor actor initiates a request for an endpoint from the endpoint directory if not already known. The endpoint response contains metadata indicating the version(s) of FHIR supported by the endpoint and any authentication and authorization requirements. After completing the necessary authentication and authorization steps, the requestor actor requests and receives the capability statement from the responder actor which includes details on the version(s) of FHIR and Implementation Guides supported.
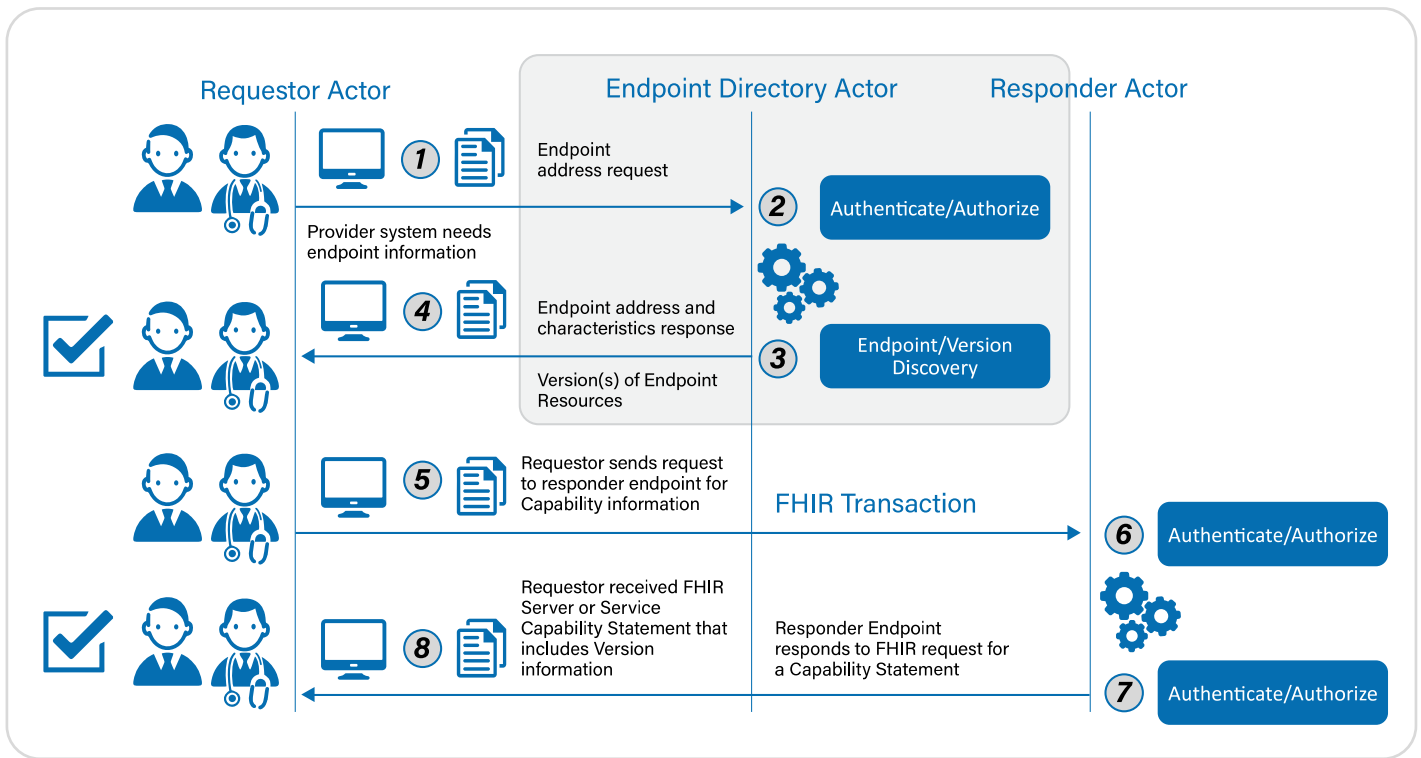
Solution 2: A US Wide Methodology for Supporting Multiple Production VERSIONS of FHIR

Next Steps and Industry Impact

SOLUTIONS

A US Wide Methodology for Supporting Multiple Production **VERSIONS** of FHIR

ARTIFACTS

Finalizing the Solution Path Forward
FAST solution documentation

INDUSTRY ENGAGEMENT

SME Session to be Scheduled

**Look for updates/results from SME Session
on the *FAST* project page**

Follow the *FAST* Technical Learning Community on LinkedIn for updates

---

The next steps for this solution are to:

Obtain SME input on the proposed solution

Next Steps

This solution will have the biggest impact on:
Payers, providers, health systems, EHRs, public health, application vendors, and intermediaries

Impacts

This solution offers these key benefits:

- Support adherence to the CMS proposed rules on reducing provider and patient burden by improving prior authorization processes and promoting patient's Electronic Access to Health Information

- Streamline and support value-based care workflows

- Increase scalability and improve interoperability between exchange partners

- Support for vendor product development and provides a venue for addressing FHIR version control issues

Key Benefits

Stakeholders of all stripes and types can help shape how this solution crystallizes for industry use by:

- Join the *FAST* Technical Learning Community on LinkedIn and stay tuned for updates coming out of the SME Session

- Visit the *FAST* project page for more information

Stakeholders