MOVING TO SDK 2

Ioannis Rousochatzakis, Publications Office, 04/12/2024

Agenda

- SDK alignment with the ePO
- SDK 2 progress & timeline
- Change management recap

EFORMS SDK – EPROCUREMENT ONTOLOGY

eNotification Standards

eForms SDK

Reception

eProcurement Ontology

Dissemination

Alignment with ePO

What
- Concepts
- Relationships
- Terminology

How
- Business entities
- Field aliases

SDK 2

Improving EFX

EFX Updates

We need

- Better visualisations
- Advanced business rules
- Better performance
- Easier maintenance

We work on

- EFX grammar
    - more expressive
    - easier to read
    - easier to write
- EFX Toolkit

Other considerations for SDK 2

- ePO alignment
  - Business Entities
  - Field aliases, TED API search fields

- ESPD Alignment
  - Generate and ESPD Request from a notice & vice versa

- Improvements suggested by eSenders

Progress update

Done

- Improvements in the EFX type system

- Pre-processor

- Improvements in EFX templates
  - Block variables
  - Variable scoping

- Support for field aliases

In progress / to-do

- Global variables and functions

- Performance improvements

- Additional built-in functions

- Pre-processor directives

Timeline

- EFX 2 grammar is on GitHub (`efx-2` branch)

- EFX toolkit 2 in alpha-4

- Work to resume early 2025

- Goals:
  - Complete the work already planned by April 2025
  - Work on ePO alignment until June 2025
  - Test internally, April - June 2025
  - Test with eSenders, Consider/implement suggested improvements (summer 2025)
  - Release within 2025

EFORMS CHANGE MANAGEMENT

Principles, methods, and safeguards

Understanding the Role of the SDK

eForms
- Specification for public procurement notices.

SDK
- Encodes and distributes the eForms specification.
- Ensures compliance and adaptability to regulatory changes.

Drivers of Change

- Regulatory changes in the specification.
- Fixes and incremental updates.
- Protocol updates for encoding improvements.

Controlling Change

- Metadata-driven paradigm for adaptive applications.
- Clear versioning system using major.minor.patch scheme.
- Incremental updates to manage impact systematically.

Frequency of Change

- Average of one maintenance release every two months.
- Gradually stabilizing to one per quarter or less.
- Major releases estimated every three years.

Adapting to Change

For maintenance releases

- Metadata-driven applications adapt automatically.

- Other applications require manual updates.

- Encouragement for gradual adoption of metadata-driven designs.

For protocol updates

- All applications need developer intervention

- Cumulative updates enable skipping versions.

- Multiple active versions support staggered adoption.

- Patching ensures older versions receive updates.

- Version lifespan

Version lifespan

- Limits the number of active SDK versions to ensure manageability.
- Allows data quality improvement at a reasonable pace.
- Compliance with changing regulatory requirements.

Communication Tools

- Machine-to-machine updates via SDK and TEDAPI.

- Release notes and SDK Explorer for version comparisons.

- Developer documentation portal and GitHub forums.

- Regular workshops for business and technical discussions.

- Helpdesk support for non-technical users.

- GitHub discussions for feedback and queries.

Closing

- Data collection
  - SDK: Flexibility and adaptability to change

- Data dissemination / reuse
  - ePO: One common vocabulary across time

THANK YOU

https://docs.ted.europa.eu/eforms-common/articles/eforms-change-management.html

Publications Office
of the European Union