2019 Annual INCOSE international workshop
Torrance, CA, USA
January 26 - 29, 2019

Bill Chown, OASIS OSLC Steering Committee Chair

OSLC Overview

Goals of OSLC

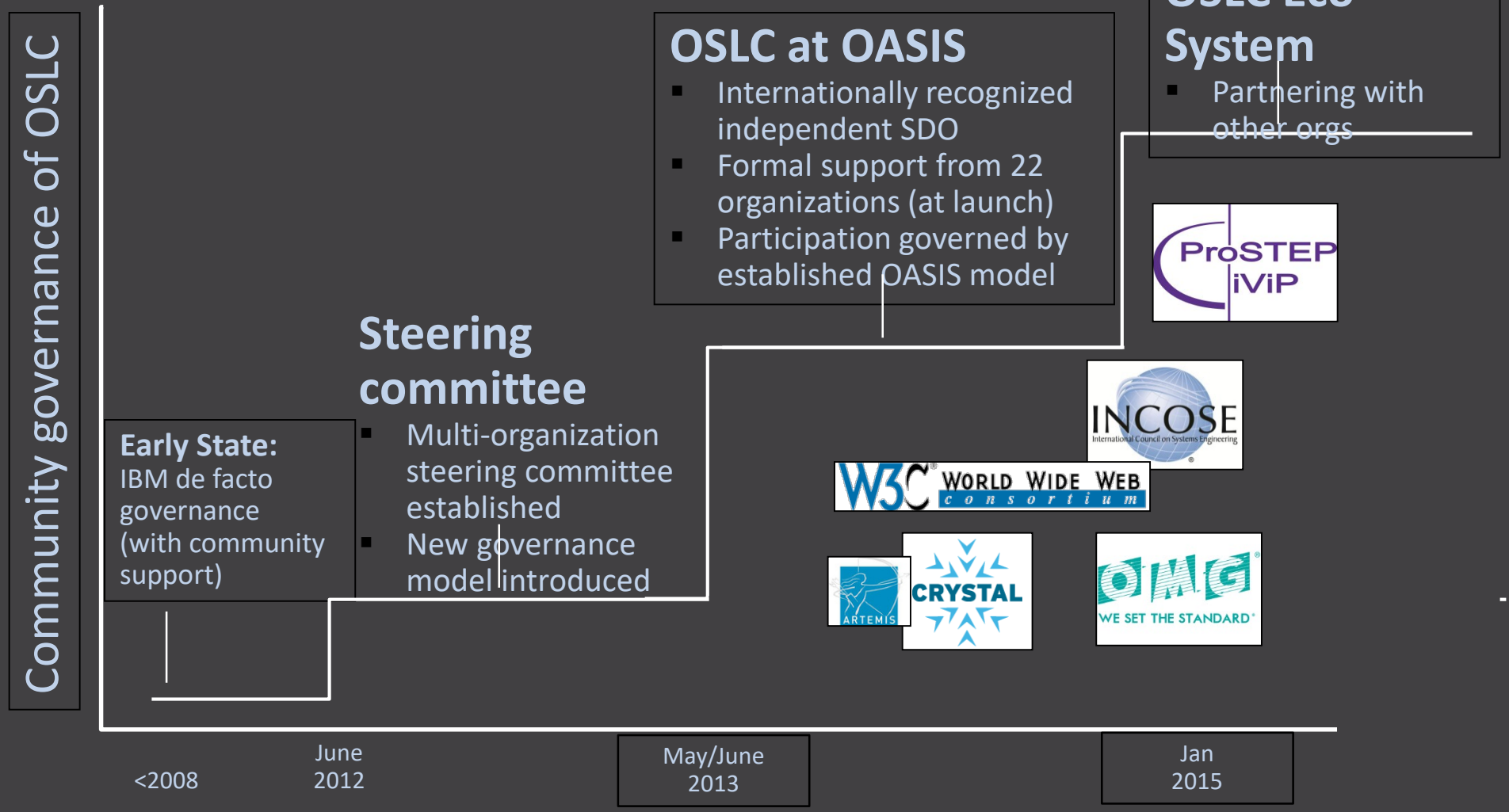
The OSLC (Open Services for Lifecycle Collaboration) initiative supports integration between a heterogeneous set of tools and components from various sources using an architecture that is minimalist, loosely coupled, and standardized.

OSLC is based on World Wide Web and Linked Data principles, such as those defined in the W3C Linked Data Platform [*LDP*], to create a cohesive set of specifications that can enable products, services, and other distributed network resources to interoperate successfully.

OSLC History

- Over 10 years old – Nov 2008
- OSLCFest in Sweden – Nov 2018
- OASIS Open – 2013
- OSLC 3.0 Specification – April 2017

OSLC Community Stepping Up



OSLC Architecture

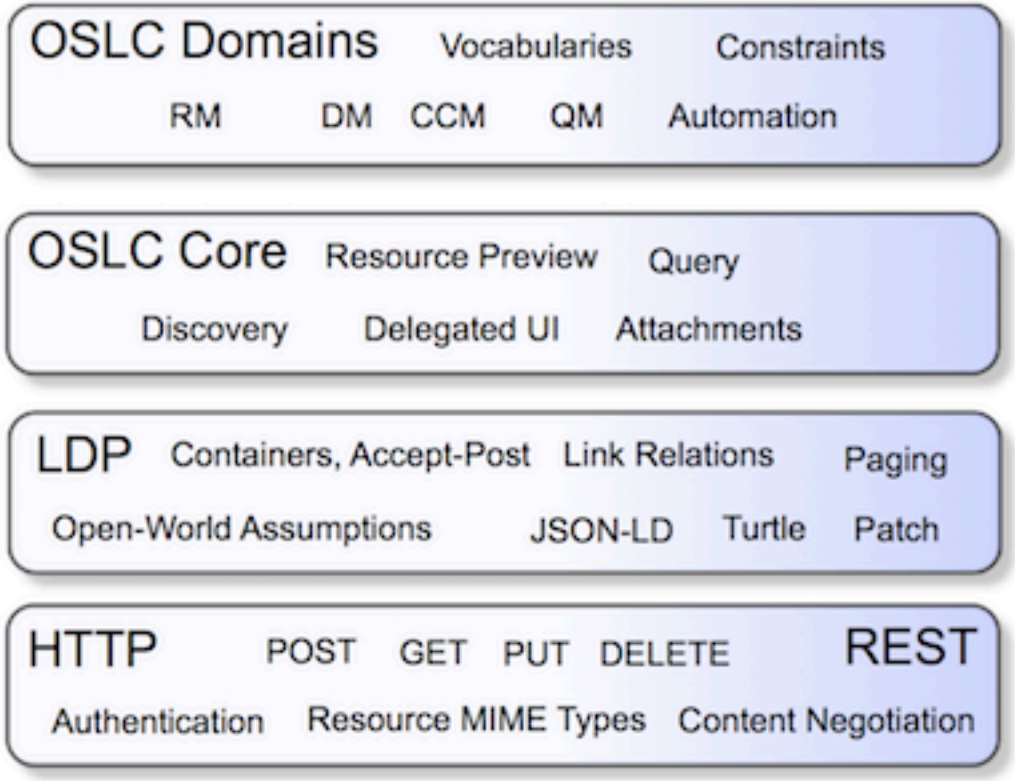
Layered architecture builds on Linked Data

Domains of interest that maintain separation of concerns and establish collaborative value streams through integration

Discoverability through Minimal, discoverable, self-describing capabilities to *enable* application integration

Reducing Variability through Self-describing, semantically rich, linked data resources leveraging HATEOAS

Address Complexity through HTTP and REST as the standard mechanism for distributed, loosely coupled APIs



OSLC Change Management 3.0 and OSLC Configuration Management 1.0 Specifications, OASIS

OSLC Core 3.0 Specification, OASIS

LDP 1.0 Specification, LDP.next Working Group, W3C

HTTP 1.1 Specification, IETF

Applicability of OSLC

- Domain-driven scenarios inspire standardization of common capabilities across disciplines
  - Disciplines include Change Management, Requirements Management, and Quality Management
  - Cross-domain scenarios such as Application Lifecycle Management (ALM) & DevOps, Product Lifecycle Management (PLM), and Integrated Service Management (ISM)

- The OSLC approach focuses on software lifecycle management to ensure it meets a core set of scenarios and requirements

Key Interactions in the Flow

- Data
  - E.g. netlist, schematic to cabling, etc. Bulk data transfer

Not the focus of OSLC today

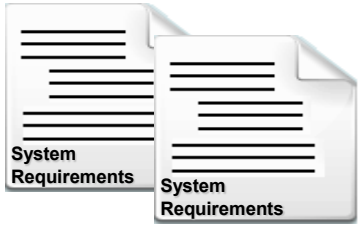
Netlist or Transform

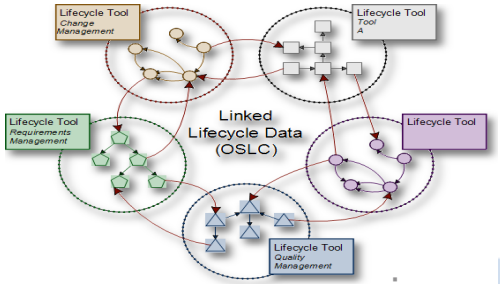
- Behavior
  - Executable models, run time code, functional co-simulation

Not the focus of OSLC at all

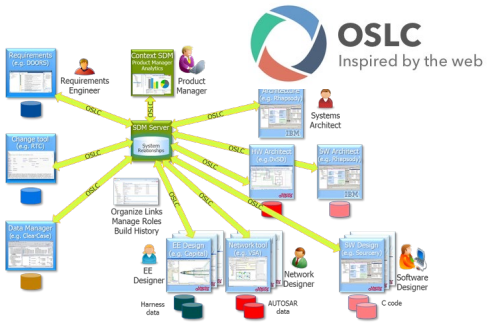
FMI, SVX, Codelink

- Intent
  - Requirements, work items, dependencies, meaning



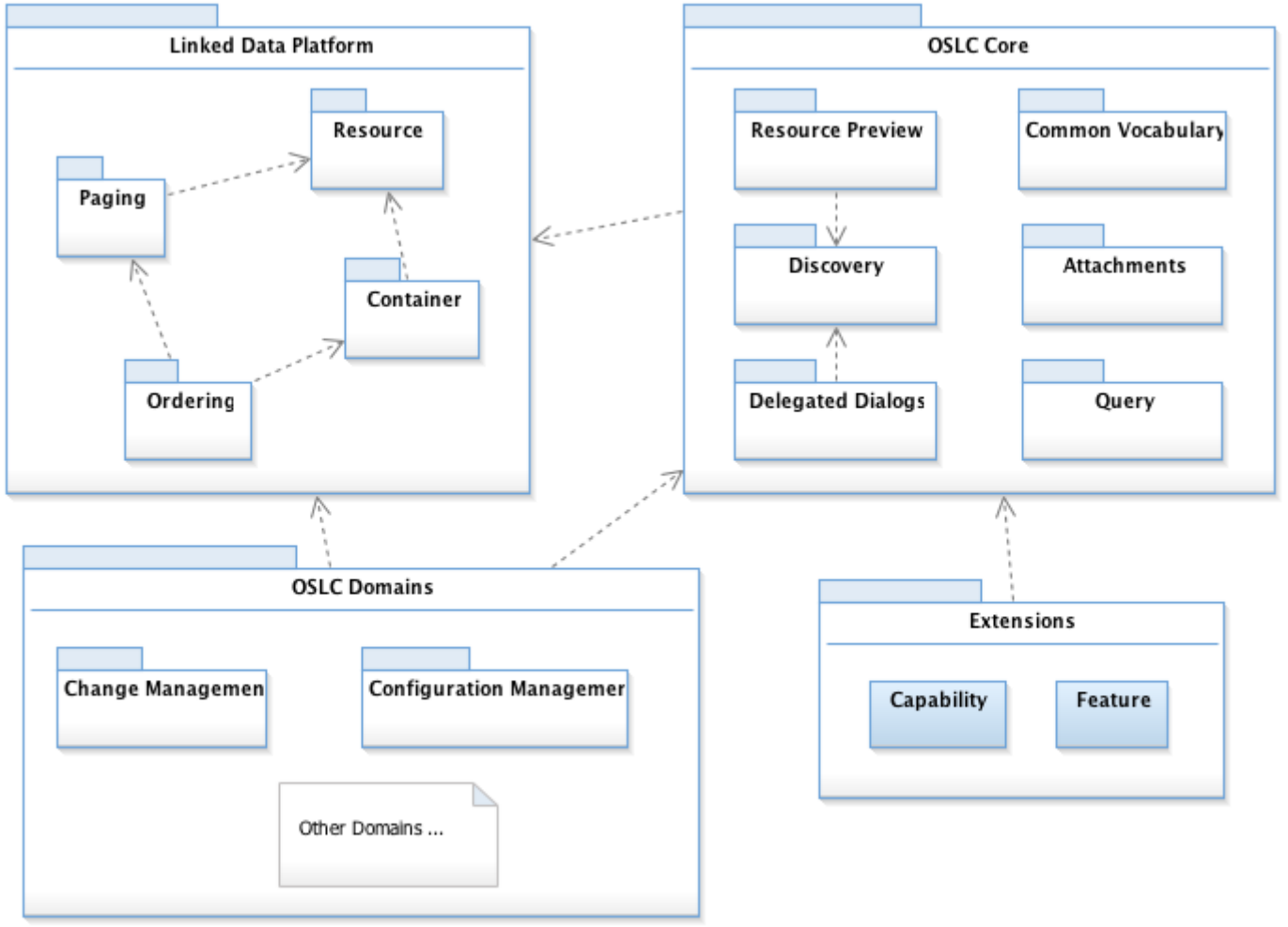


Linked Data, Shared Information



OSLC Core 3.0

- OSLC Core 3.0 builds on capabilities developed in different standards organizations, TCs and working groups



Goals of OSLC Core 3.0

- Integration is based on an open standard, and not controlled by any single vendor
- OSLC 3.0 is based on the new W3C Linked Data Platform standard which provides a solid foundation for reading and writing linked data resources
- The specifications are simpler, more consistent and will potentially be more attractive to, and easier to consume by new integrations
- There are some new capabilities specified, including Attachments, inverse link labels, traceability and impact types
- Domain vocabularies can be improved for data consistency and removing data gaps
- All Resource Shapes are provided in machine readable *Turtle* files

OSLC Domains

A new Domains WG (TC) is formed within OASIS-OSLC

- Combined attention to the various domains, rather than separate domain groups
- Migrate existing finalized OSLC v2.0 domain specifications
  - Tracked Resource Set 2.0,
  - Architecture Management 2.0
  - Asset Management 2.0
  - Automation 2.1
  - Performance Monitoring 2.0
  - Quality Management 2.0
  - Requirements Management 2.0
- Capture requirements and use cases for other related domains

Achieving the Digital Thread

Use OSLC to connect your data and achieve the digital thread across domains, applications, and organizations

2019
Annual INCOSE
international workshop
Torrance, CA, USA
January 26 - 29, 2019

www.incose.org/IW2019