



## Data-as-a-Service Design Pattern Public Forum Office of Technology Strategies September 16, 2014

### Meeting Summary

**Purpose:** The purpose of this Public Forum was to provide an opportunity for VA and external experts and stakeholders to learn about the first increment of the Data-as-a-Service (DaaS) enterprise design pattern and provide feedback prior to the Design Pattern's finalization, publication and implementation.

**Background:** The Technology Strategies Design Patterns Team has been soliciting input for the development of this DaaS Design Pattern. This involved collaboration efforts with a variety of stakeholders, including internal VA subject matter experts (SMEs), external government SMEs, industry vendors, and members of academia. This Public Forum represented the final stage of stakeholder engagement in the Design Pattern development process.

**Overview:** The TS Design Pattern Team presented the final draft of this increment of the DaaS Design Pattern. An introduction of the department was provided by Caitlin Ellsworth followed by an overview presented by Jacqueline Meadows-Stokes, the Government lead for the DaaS Design Pattern from the Office of Technology Strategies (TS). Justin Ronkowitz of the Design Patterns Team then presented the draft increment of the Design Pattern and covered the details of its content and development. Stakeholders from various vendor companies and internal and external stakeholder organizations, shared their questions, comments, and insights related to the content covered in the presentation.

#### Key Discussion Points:

The presentation content can be found in Appendix B.

Key items discussed during the Q&A portion of the public forum are paraphrased and summarized below.

- **Centralized vs. de-centralized data sources**
  - Service-Oriented Architecture (SOA) enterprise services often provide de-centralized but consistent enterprise data stores.
  - Will the services presented in the design pattern only work within a centralized VA-enterprise data structure?
    - Whether centralized or de-centralized, there is still a need to define authoritative data sources, which can then be used to offer the data as a service.
  - What is implied by centralized data sources?
    - Data does not have to be consolidated on a service data layer, but the shared services layer is still a centralized platform from where the data is accessed, with the data itself is being accessed from all across the enterprise. This will be clarified within the final version of the design pattern.

- **DaaS as a data aggregation service**
  - The DaaS design pattern may result in development of a data source(s) to aggregate desired data to be sent back to the application user.
  - There is still a domain-level service which determines data sources and an aggregator layer which collects and distributes data back to the user at the application level.
- **Data governance for the aggregator to identify underlying data sources**
  - All data requests go through the EMI and the data aggregator, with the exception of bypass permissions around the EMI, which have to be defined and finalized as a data policy to determine when a request for data aggregation and access may bypass the EMI.
- **DaaS revisions as part of Enterprise SOA level design pattern**
  - Disclaimers will be noted that this design pattern is part of a previous version of the Enterprise SOA design pattern, and models/diagrams presented will be updated as part of the SOA enterprise framework being continually developed as part of the SOA enterprise design pattern.
- **Flexibility with enterprise orchestration and choreographies**
  - Authentication and authorization may have been addressed by previous VA enterprise orchestration layers before reaching the data layers. But for this stand-alone DaaS design pattern, the authentication and authorization will be replicated by the DaaS as the user access enterprise data stores.
  - The enterprise SOA Design Pattern update will eventually include guidance and design inputs to approach shared services across all applications, including data access and aggregation at the orchestration level.

Key items discussed during the outstanding issues and final thoughts portion of the public forum are paraphrased and summarized below.

- **Data and information security and compliance**
  - VA-6500 handbook requires that all data at rest must be encrypted, however it is still being clarified whether the current version of VA-6500 requires encryption at the data base level or the disk/storage level.
  - The current update for the VA-6500 is clarifying that the data must be encrypted at the data base level.
  - The design pattern will reference VA-6500 in order to denote that current VA data encryption design will always be compliant with information security protocols as updated in VA-6500.
  - Based on future VA-6500 versions, performance and availability of data at rest within the DaaS design patterns will be addressed after the next VA-6500 version due out in September.
- **Design pattern implementation**

- For performance targets and SLAs related to enterprise shared services (ESS), it is currently beyond the scope of this document to define performance and availability requirements and targets. These will be handled at the development and implementation level, related to the adoption and use of the ESS.

**Next Steps:**

1. The TS team will finalize the Design Pattern for VAIQ submission and formal review. Updates will include:
  - a. Changes to remove confusion associated with centralized vs. de-centralized data stores
  - b. Additional language to explain the “To-Be” design pattern and the need for future updates based upon development of the Enterprise SOA Design Pattern document(s)
  - c. Changes in Data Security section language to reference VA-6500 in order to denote that current VA data encryption design will always be compliant with information security protocols as updated in VA policy
  - d. Finalization of use cases to be more concise and focused on specific VA data calls to identified DBMSs
2. The TS team will evaluate feedback from the development of this Design Pattern to incorporate into the final version and the next increment of the overarching DaaS Design Pattern, as well as provide inputs into further iterations of the SOA Design Pattern.

**Appendices:**

- A. Participant List
- B. Presentation Slide Deck

## Appendix A: Attendee List

<b>Last Name</b>	<b>First Name</b>	<b>Affiliation</b>
Allison	Michael	VA
Badami	Kash	MarkLogic
Baggs	Marcus	VA
Barrineau	Dirk	VA
Behr	Steven	BAH
Bertin	Dave	Red Hat
Bogden	Nicholas	VA - TS
Brooks	Joseph	VA - TS
Burke	John	VA - OIT
Butler	John	Everware
Cahall	Timothy	SBG
Cox	Keith	VA
Cronkite	Wesley	MPS
Cunningham	Robert	VA
Dance	Michael	BAH
Das	Upendra	PwC
Davis	Michelle	Red Hat
Dinkle	Chris	Oracle
Divi	Kamal	MITRE
Doane	Michael	MarkLogic
Ellsworth	Caitlin	MPS
Emery	Rodney	VA - TS
Fegley	Matt	VA - EA
Feldman	Damon	MarkLogic
Galway	Kim	MITRE
Garland	Trisia	VA
Gould	Stephen	PwC
Grant	Walter	JWB
Hopler	Douglas	ACET
Huddleston	Michael	BAH
Imam	Shad	Oracle
Islam	Tariq	Oracle
Kerr	James	MongoDB
Khalil	Ambreen	BAH
Khalsa	Gurujeet	
LaForest	Will	MongoDB
Laskey	Kenneth	MITRE
Luedkte	Terry	VA - PPM
Mandala	John	

<b>Last Name</b>	<b>First Name</b>	<b>Affiliation</b>
Marks	Mary	VA – Bay Pines
May	Thomas	PwC
Meadows-Stokes	Jaqueline	VA - TS
Memon	Afzal	Oracle
Minnick	Larry	SBG
Mulford	Carrie	
Nitti	Luke	MITRE
Nixdorf	Jason	Amazon WS
Odunlami	Adeola	ACET
Oestreicher	Donald	7Delta
Orifici	Robert	VA
Patel	Anil	AgileSDE
Peeples	Matthew	Amazon WS
Peterson	Dennis	VA
Roach	Dennis	BITS
Ronkowitz	Justin	BAH
Sastry	Anand	Engility
Saxena	Ritesh	IBM
Scheich	Chris	MongoDB
Shinozaki	Michael	Microsoft
Tirrell	John	VA - OIT
Vogel	Skip	Oracle
Waters	Brandon	Waters
Williams	Shelley	Harris
Wronko	Maryann	
Yannakopoulos	Demetrious	