

Make Spring Home *(customization & extensibility)*

Sasi Peri

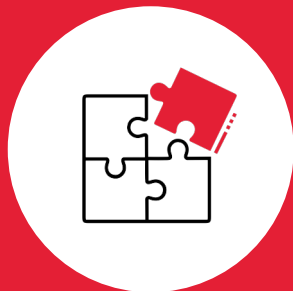
We are *Essential to care*TM



We deliver products and solutions to **improve the lives of people every day**



We power our
customers' missions



By bringing together
**unmatched solutions from
across our enterprise**



To achieve both **clinical
and operational success**

Our scale, breadth and expertise maximize your impact

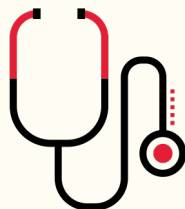
A GLOBAL, INTEGRATED HEALTHCARE PRODUCTS & SERVICES COMPANY



~**90%**
of U.S. hospitals



>**29k**
pharmacies



>**10k**
specialty physician
offices and clinics



3M+
patients



>**6.5k**
labs

46k+
home healthcare
products

>**51k**
lab products

Safe Harbor Statement

The following is intended to outline the general direction of my findings and is intended for information purposes only. You are encouraged to do your own thorough analysis and research before the use of the technology in your production grade applications.

Agenda

1. Why am I doing this presentation?
2. Demo & Code
3. Watch out !
4. Reference links, source code (GIT)
5. Q & A

Why and what?

In this session I will demonstrate spring's extensibility and customization capabilities that can help us create reusable **custom starters** (*to scaffold commonly used business functionality*) and create **custom developer experience** to consume the same (*standard and easy means to consume the same*). There are many more aspects that can be customized, extended, but in this session specifically I will demonstrate

1. Custom starters (alexa skill boot starter)
2. Custom initializr
3. Putting both together and live code (build live skill)
4. Add some security (account linking)

Custom Starters

(alex-spring-boot-starter)

Why custom starter?

1. Custom reusable functionality packaged into libraries (hide complex and common logic)
2. Scaffolds these commonly used custom and 3rd party libraries & manage dependencies.
3. Auto configures as needed.

Alexa Sprig Boot Starter?

Jump starter to host a Custom Skill as a Web Service, using SpringBoot

ALL ROADS LEAD TO

Spring Initializr



Creating 2nd best place on the “intranet”
(Custom Initializr)

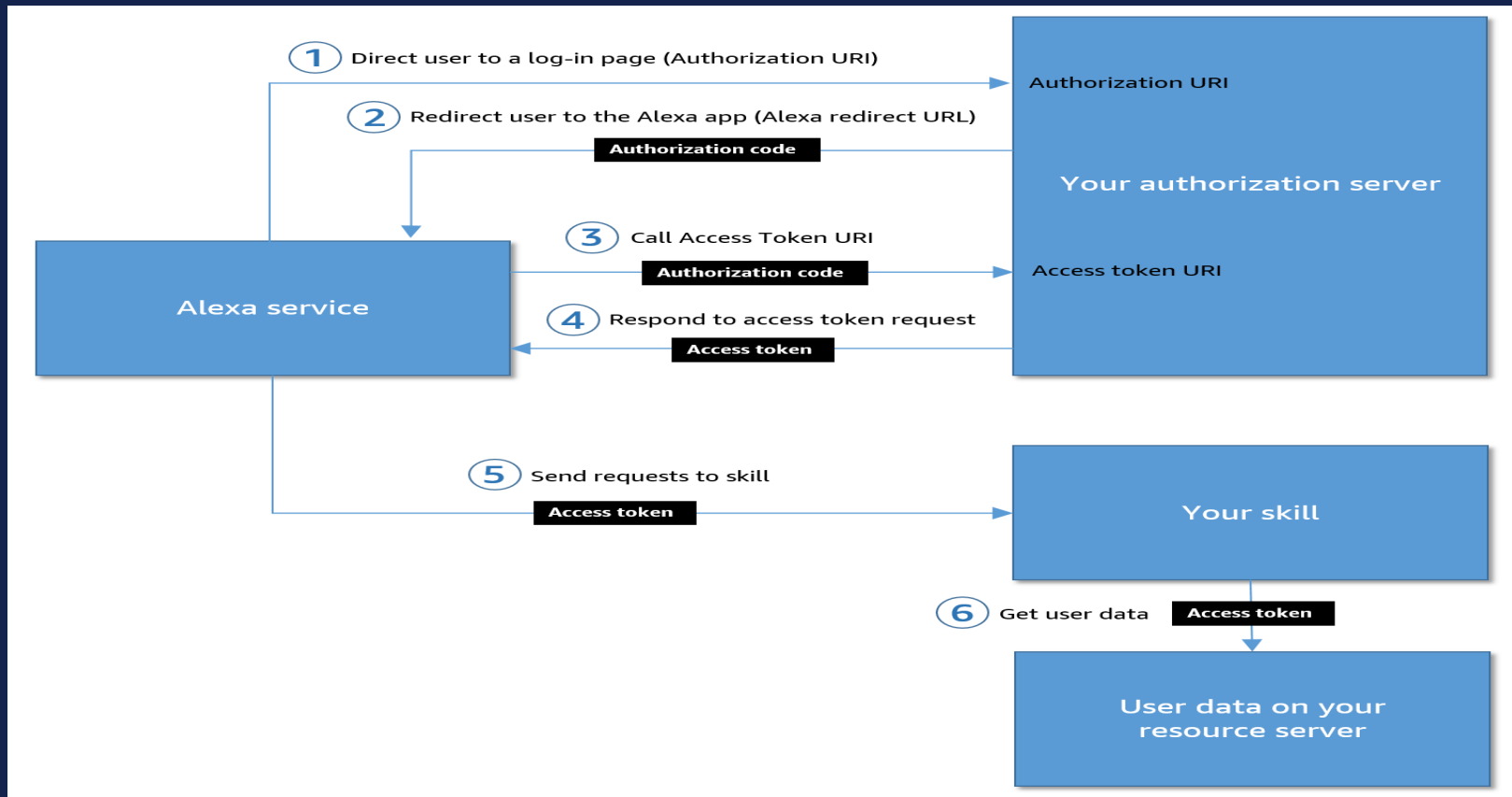
Why custom initializr?

1. Create common developer experience to consume
 - ✓ Add custom libraries, starters via starter POM.
 - ✓ Jump start the development
 - ✓ Specific needs such as templates pre-populated with your company's standard tools and libraries
2. Network, fire wall issues, that may prevent

Build Sample Skill

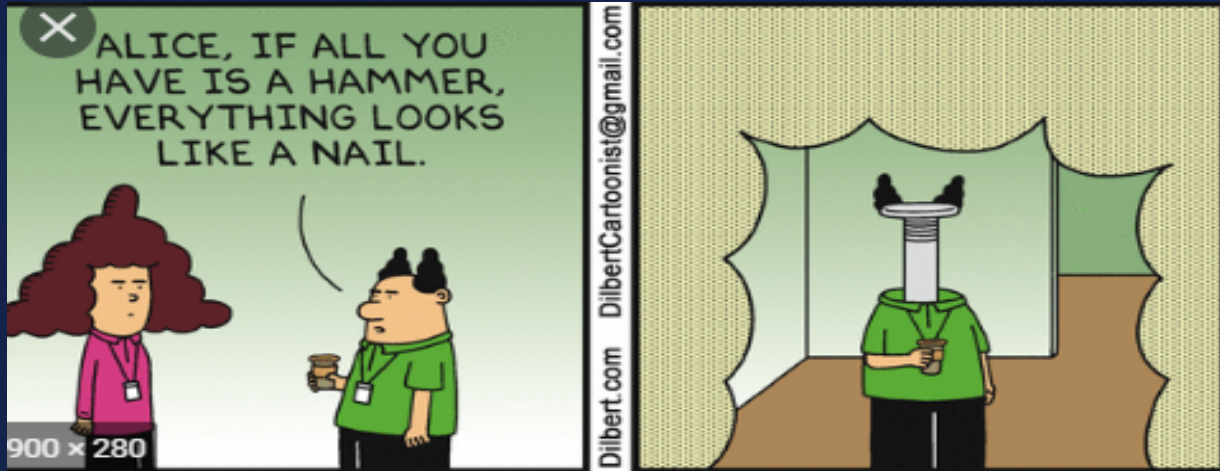
Account Linking - Demo

Behind the scenes



Before we end !

Beware



Beware



Standards naming conventions and more

Links and References

Demo source and documentation

- ✓ Source: <https://github.com/sasiperi/alexa-spring-boot>
- ✓ Maven-repo: <https://mvnrepository.com/search?q=io.github.sasiperi>
- ✓ Blog/Documentation:
 - ✓ <https://sasiperi.github.io/>
 - ✓ <https://sasiperi.github.io//blog>

Others

- ✓ Initializr:
 - ✓ <https://docs.spring.io/initializr/docs/current/reference/html/#create-instance-boms>
 - ✓ <https://medium.com/digitalfrontiers/how-to-customize-the-spring-initializr-2439ecabb069>
- ✓ Starter:
 - ✓ <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/#boot-features-custom-starter-naming>
- ✓ Amazon/Alexa:
 - ✓ <https://developer.amazon.com/en-US/docs/alexa/custom-skills/speech-synthesis-markup-language-ssml-reference.html?&&&>
 - ✓ <https://developer.amazon.com/en-US/docs/alexa/account-linking/account-linking-concepts.html>

Similar Sessions

- ✓ Unleash Power of Spring Cloud: <https://springone.io/2020/sessions/unleash-the-true-power-of-spring-cloud-learn-how-to-customize-spring-cloud>

Thank you

1. Jim Shingler – Director, Digital Transformation and DevOps Leader
2. Ralph Meira – Advisory Platform Architect, Vmware
3. Brian Holdren – My friend at LeanDog

Spring Has Sprung

Q & A TIME

