Native Applications WG (NAPPS)
May 5/14

Paul Madsen, Ping

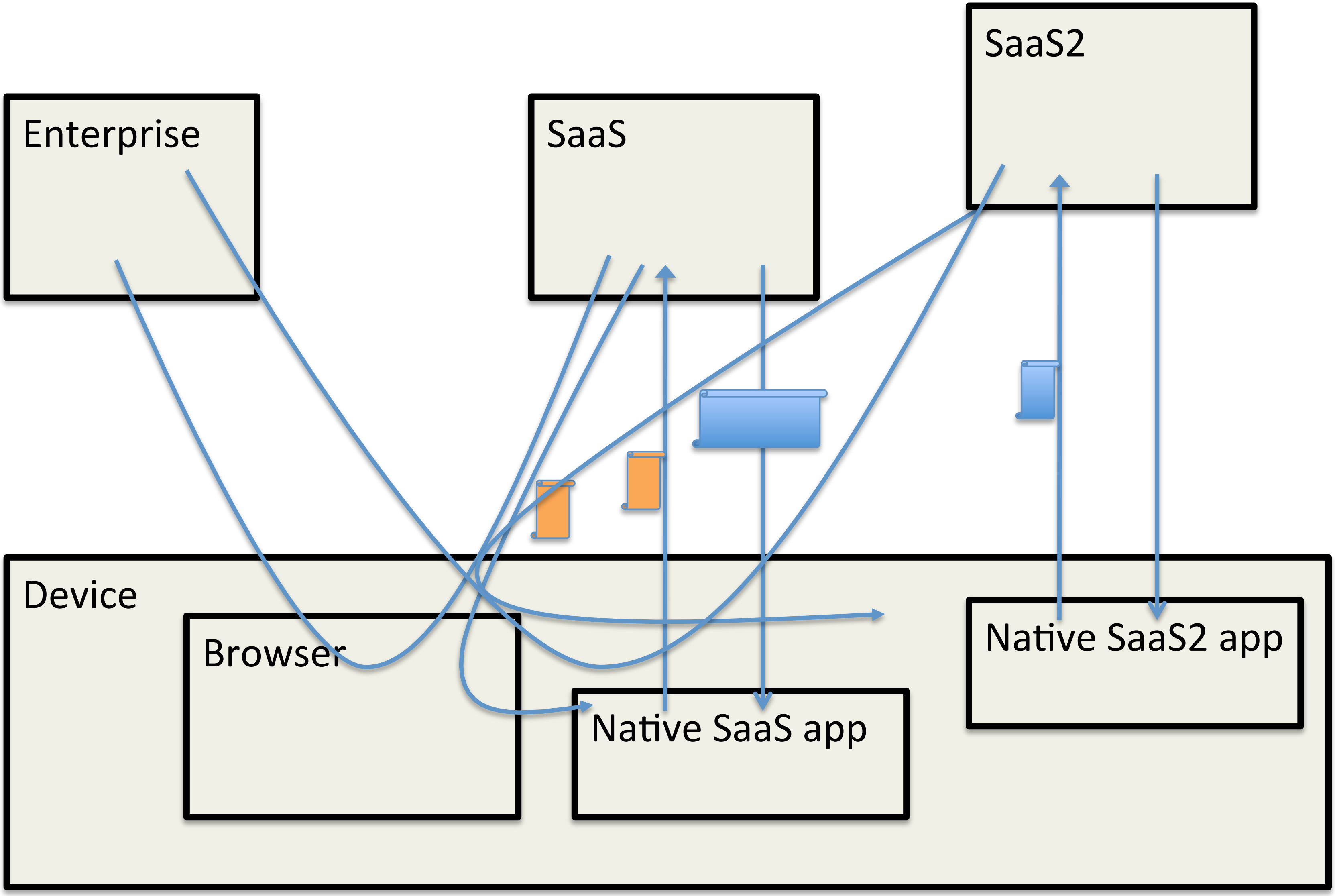
NAPPS value proposition

- OAuth & Connect enable native applications calling APIs
- Popularity of native application model has led to an explosion – enterprise employee might have over 20 different apps
- Neither, out of the box can enable a SSO experience across native applications
- Even with refresh, not insignificant usability burden

Default pattern for SaaS native

- Native application uses OAuth to obtain tokens to be used on API calls
- Employee authenticates to/authorizes each application individually
- Employee interacts with each OAuth AS (corresponding to each API) to obtain an OAuth token
- Best practice is to use federated SSO from the enterprise into each SaaS

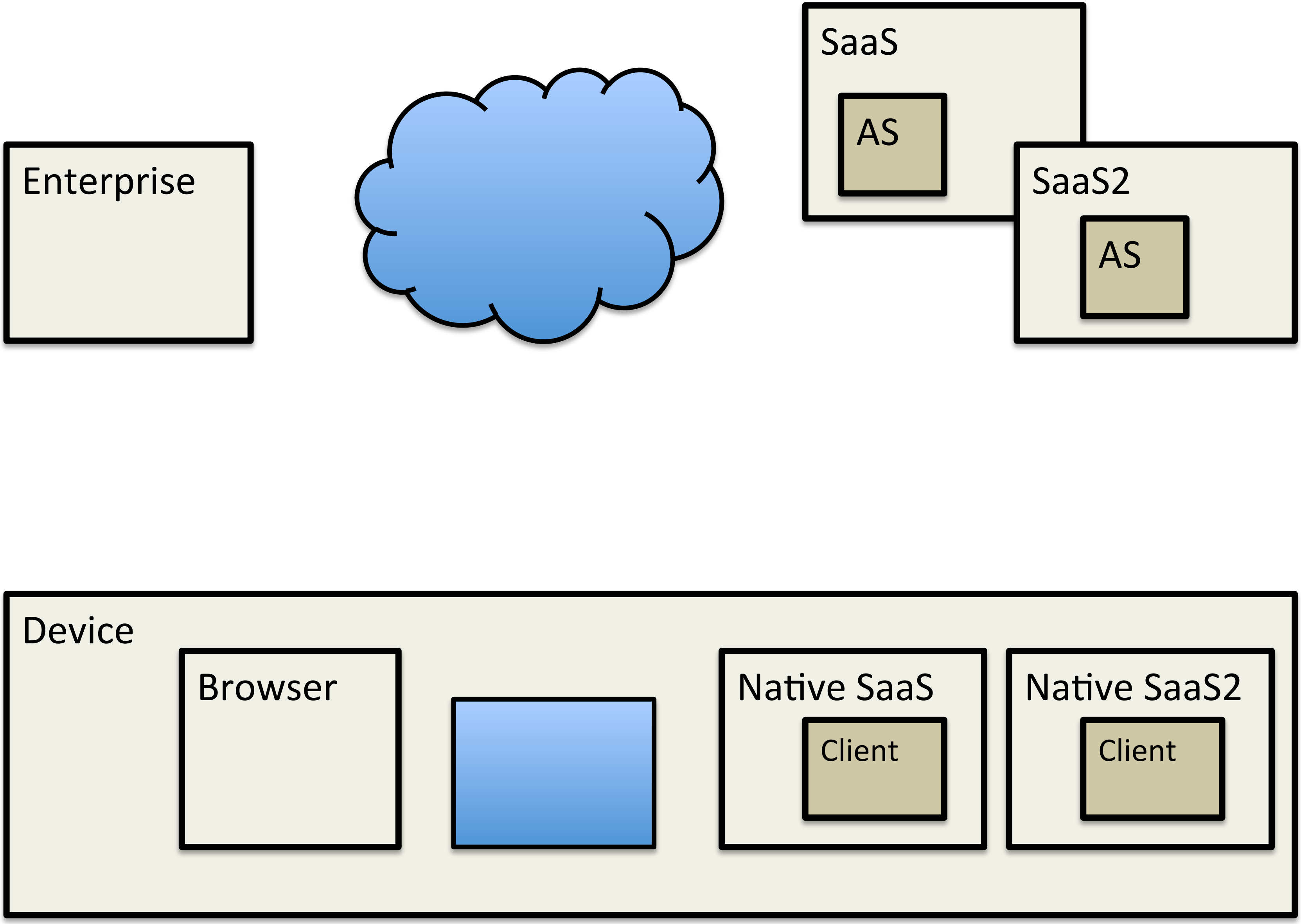
Default flow



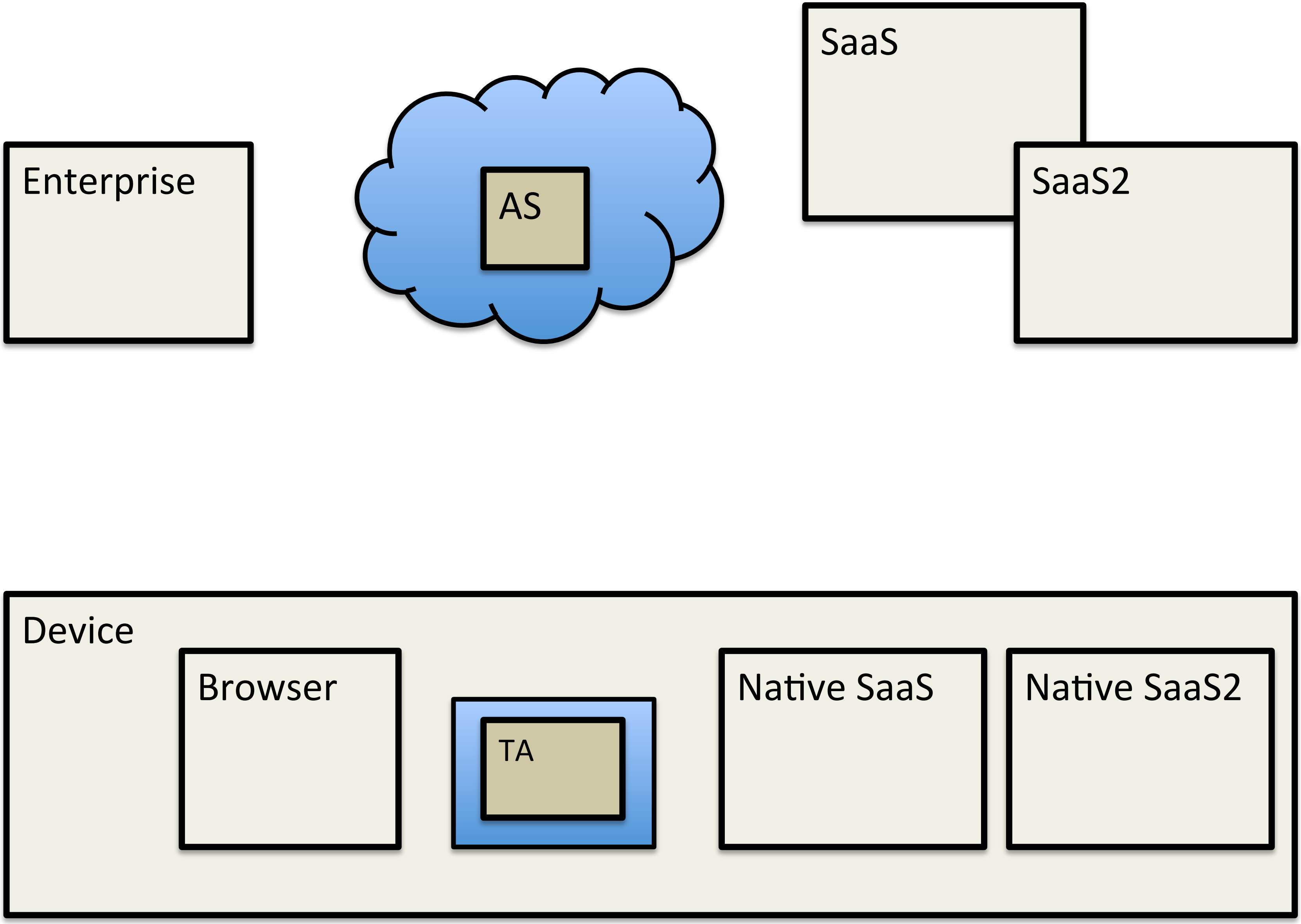
Implications of default pattern

- Employee bears burden of authenticating/ authorizing each native application separately
- Even if done infrequently, may be unacceptable
- Each SaaS must directly support OAuth (running an Authorization Server)
- Enterprise removed from employee's use of native applications

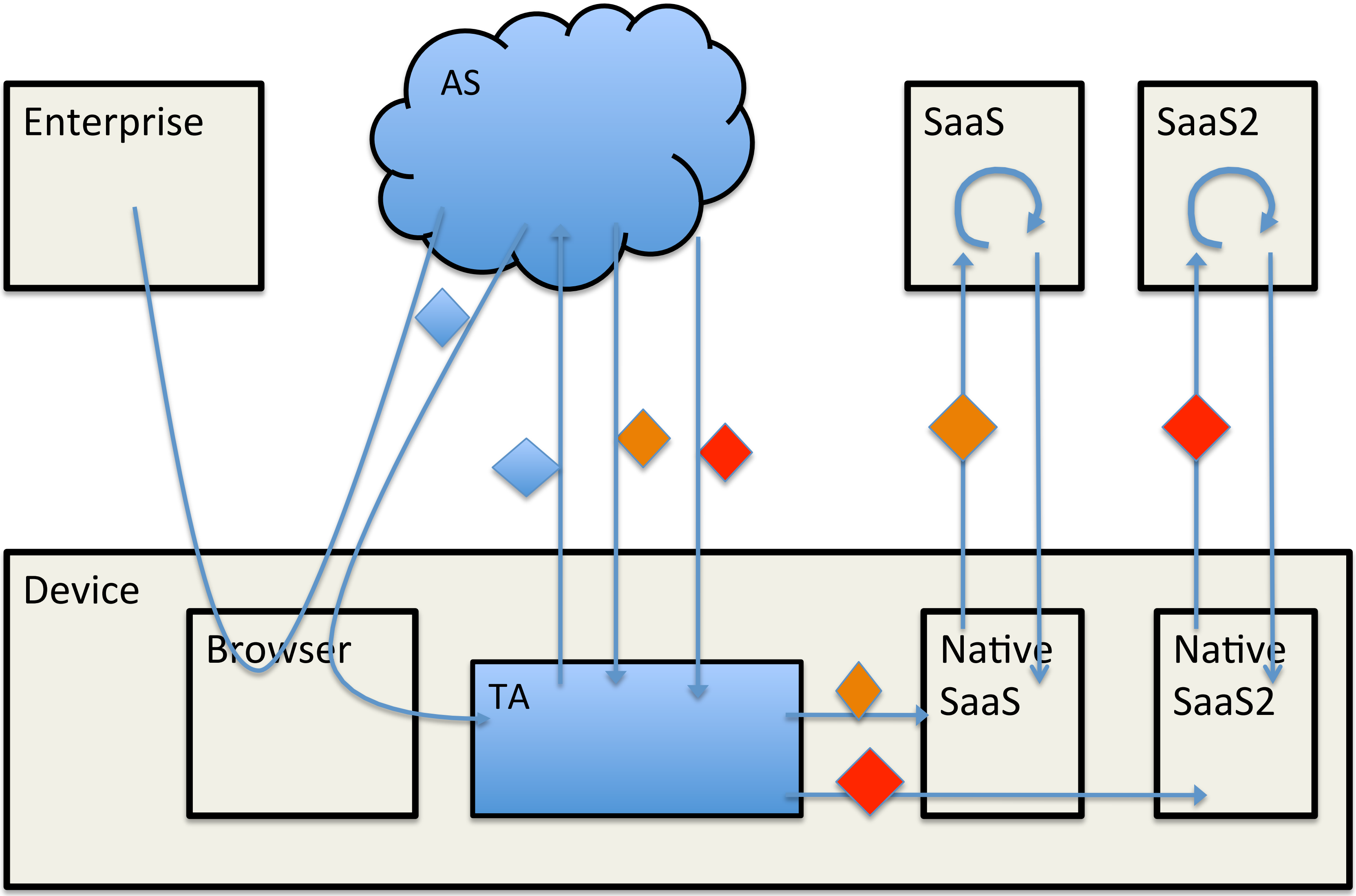
Alternative



Alternative



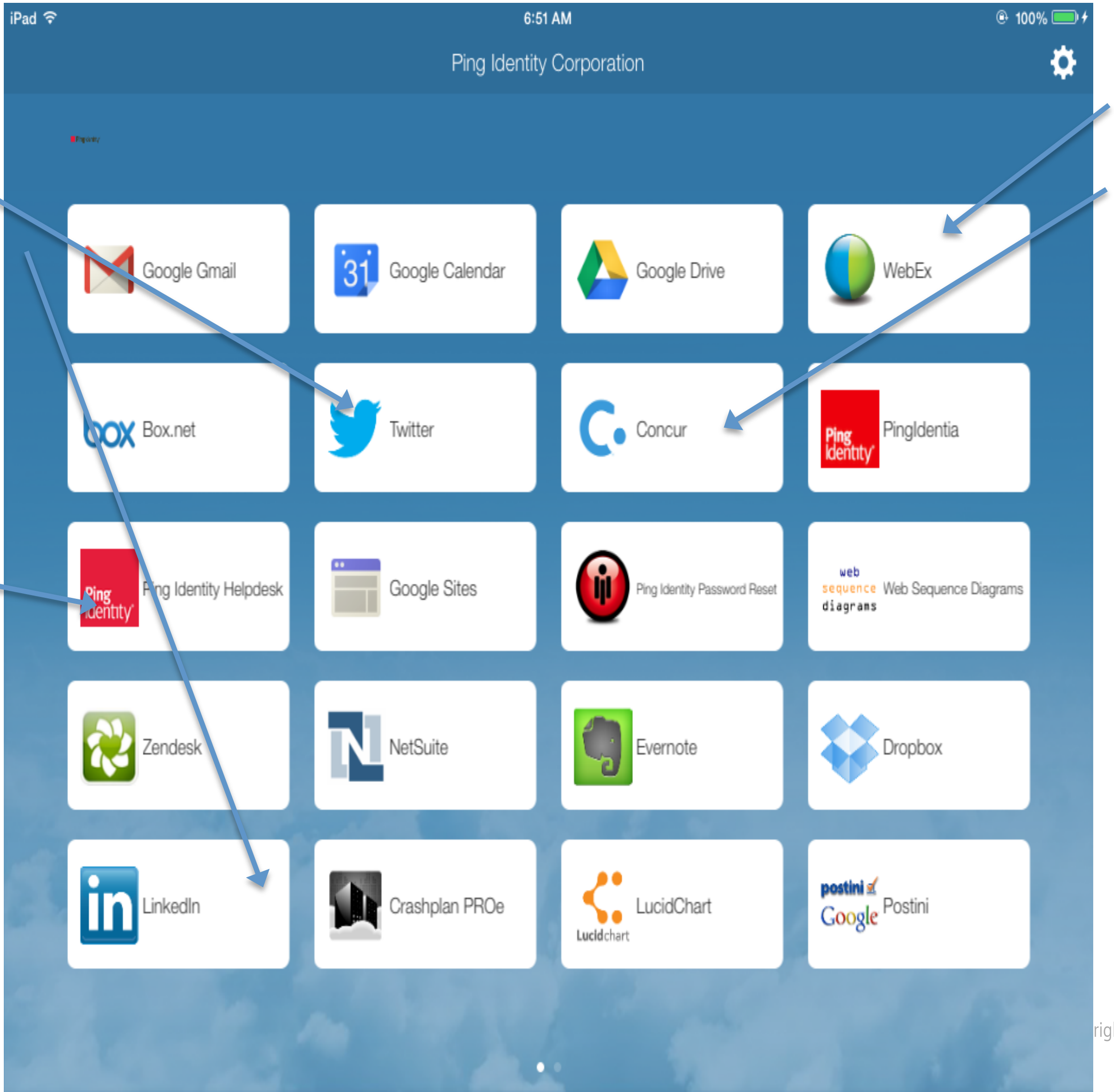
Alternative



Advantages

- Employee performs explicit authentication & authorization only for the TA – results in tokens issued down to the TA like any OAuth/OIDC client
- Other apps able to benefit from this TA authentication for their own – TA tokens used to obtain application tokens
- User can enjoy SSO across those native applications

Launcher UX



Vaulted Web

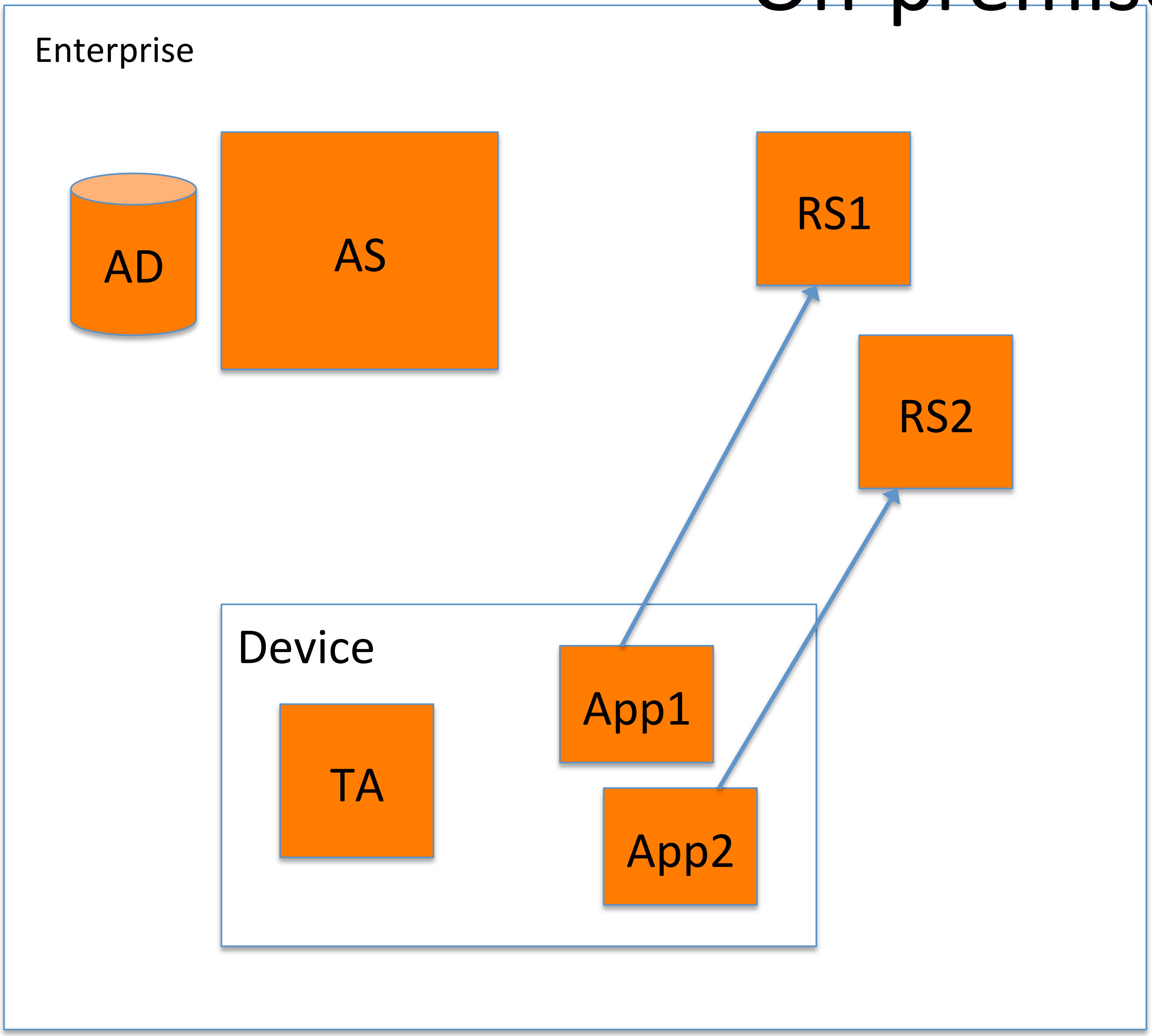
Federated Web

Native

What plays Token Agent role?

- Dedicated SSO application
- Business application
  - Peer model
- MAM Agent
  - Eg MobileIron, Airwatch , Good etc
- OS?
  - Samsung KNOX
  - Android Account Manager, Google Play etc

On-premise



Enterprise

AD

AS

RS1

RS2

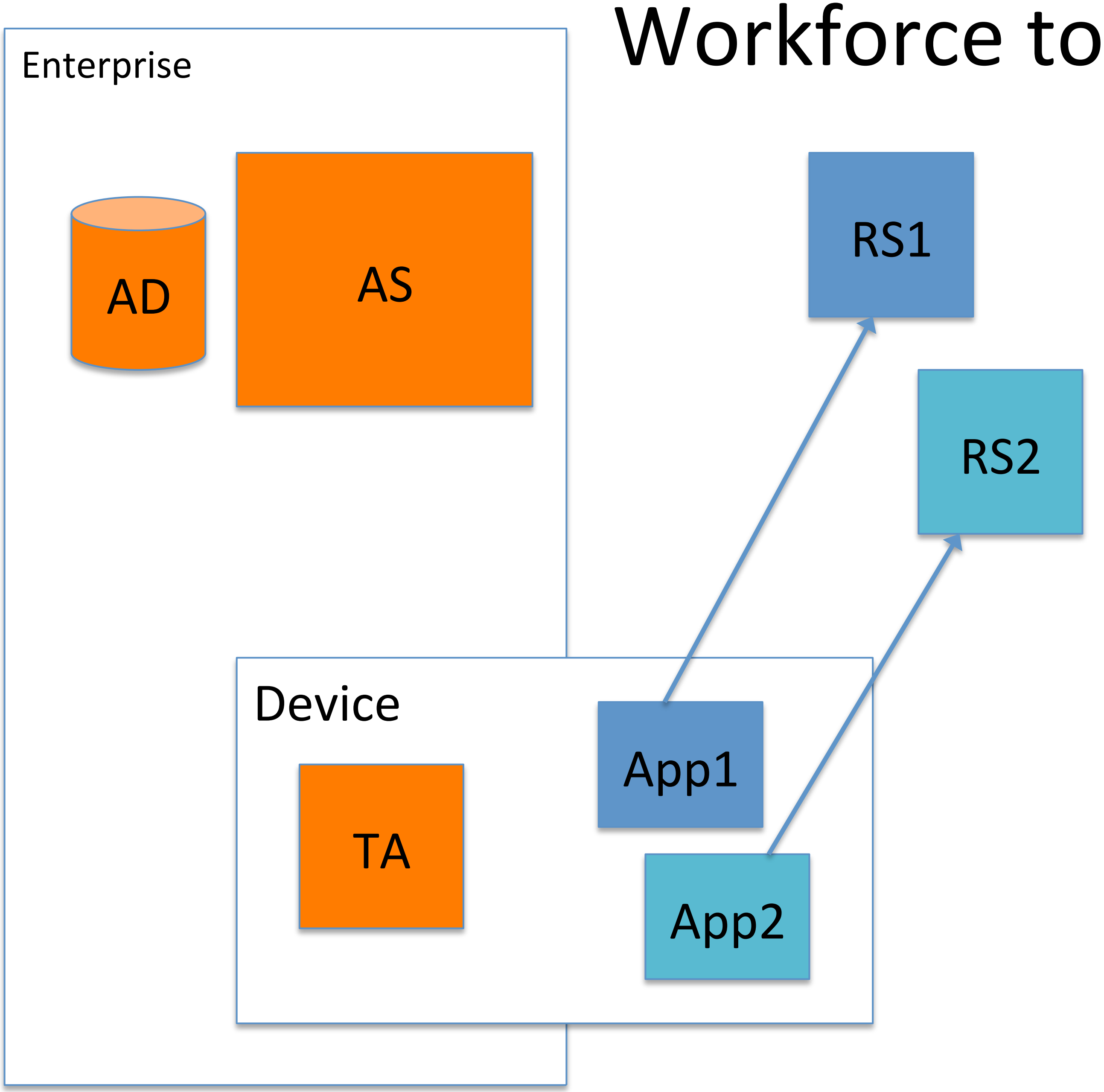
Device

TA

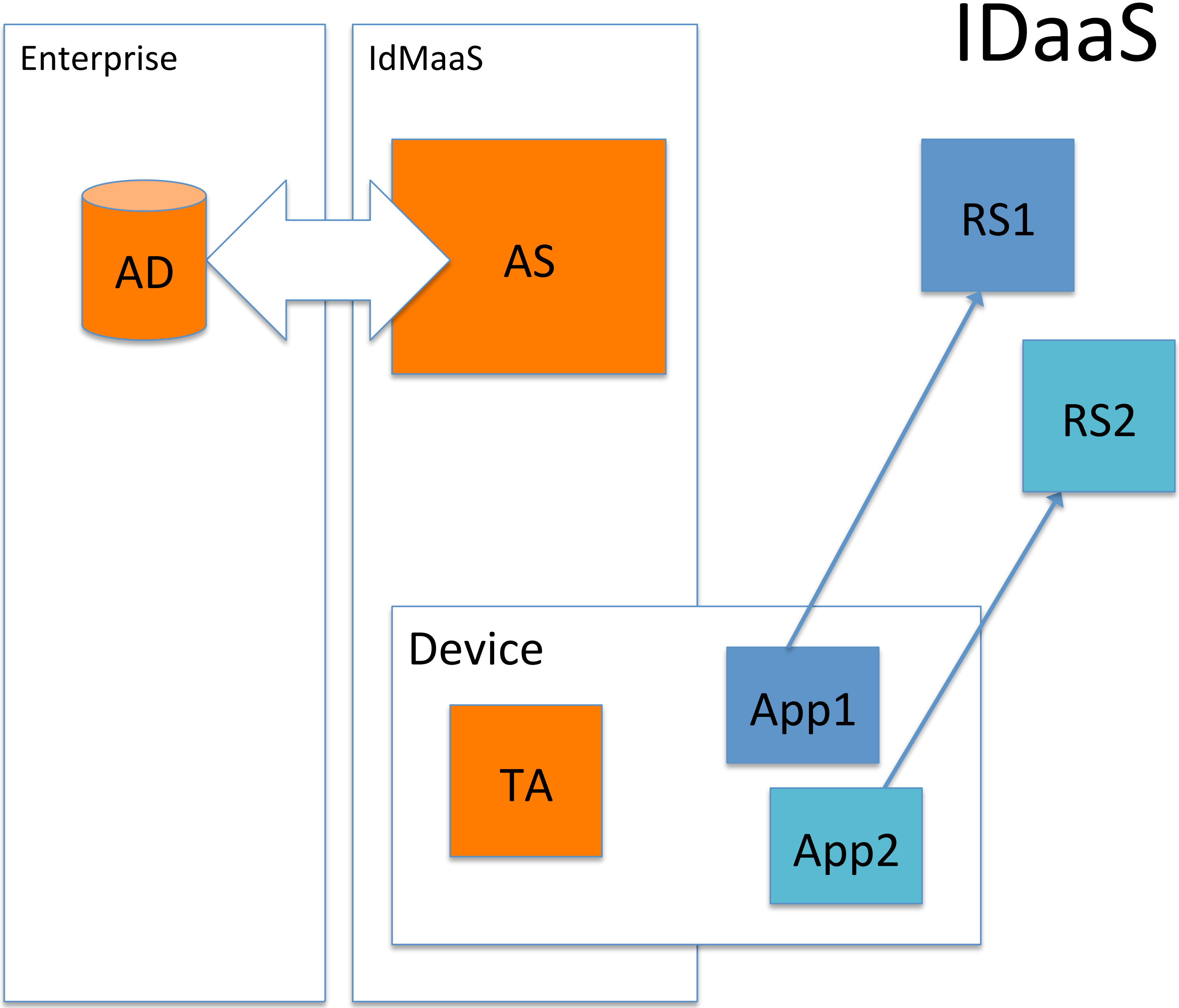
App1

App2

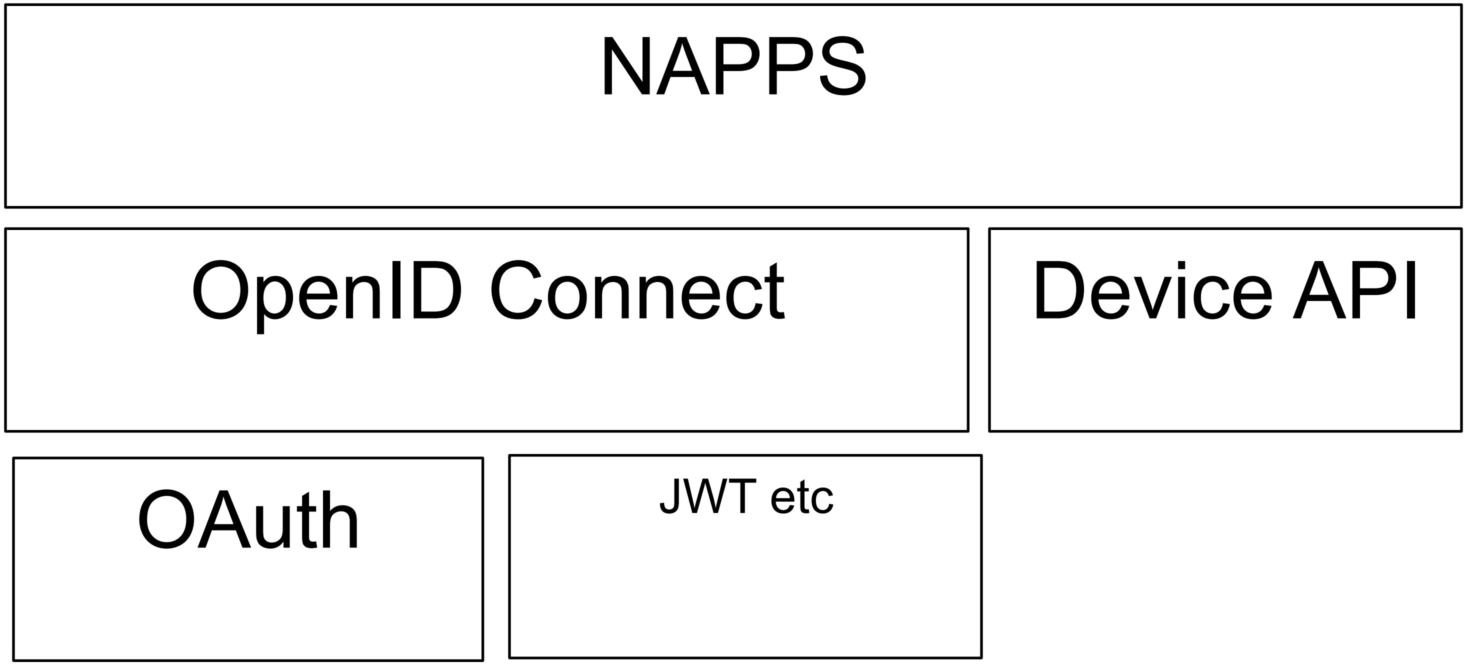
Workforce to SaaS



IDaaS



Stack



NAPPS

OpenID Connect

Device API

OAuth

JWT etc

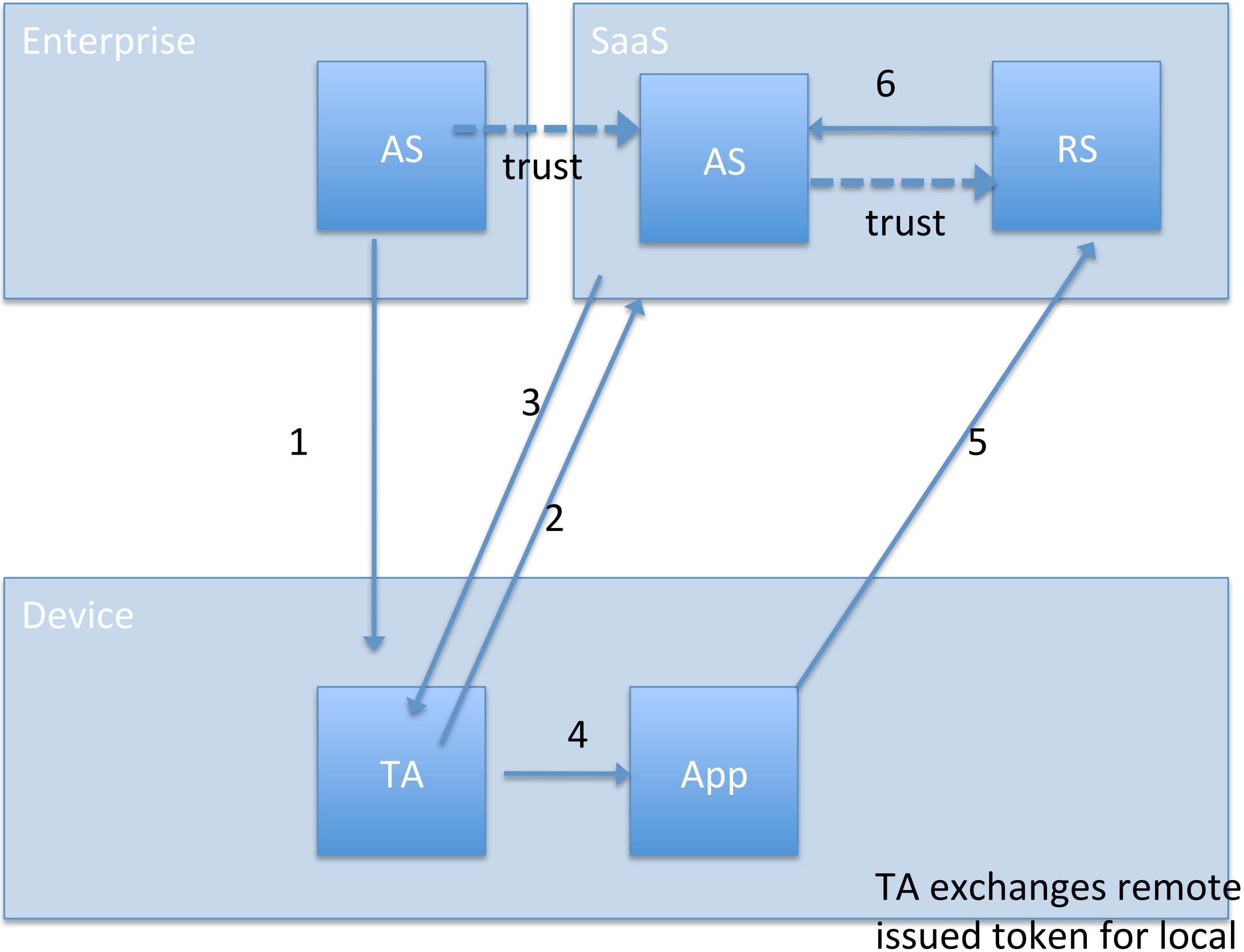
Current model

- TA authenticates the user at ‘Home AS’
- TA obtains its own refresh, access & id_token
- Uses access token to call AppInfo API for app metadata
  - Icons, custom URL scheme, etc
- Uses refresh token to (when relevant) to request
  - a secondary access token for a particular native app
  - An id_token targeted at a Remote AS that can be exchanged for a secondary access token for a particular native app

Federation?

- If the AS issuing tokens to the TA is in a different policy domain than the RS then we need mechanisms/protocols to cross that policy & trust boundary
- Demands a federated model

Federation – Burden on TA



TA exchanges remote AS issued token for local token

Consent?

- The Remote AS may want to collect its own consent for its set of RSs
- Current proposal is that TA send the user agent to that AS authz endpoint in authenticated state, ie web SSO from the Home AS
- Ongoing discussion as to best way to accomplish this

Status

- NAPPS WG has calls bi-weekly
- Reasonably active mailing list
- Good membership across IdM vendors, large SaaS, interest from MAM
- Two specifications progressing
  - Core (John Bradley, Ping editing)
  - OS Bindings (Thomas Debenning, OneLogin)
- Hoping to have an Implementors Draft for July time frame (and so possibly an interop for Cloud Identity Summit)
- Will meet at IIW

Implementations

- Multiple implications of an ‘Authorization Agent’ (AZA) exist (Ping, VMWare, MobileIron, and more) that predates NAPPS
- Lots of logically similar functionality (Google Play Services, Layer7, Centrify, Adobe)
- Corresponding server support in Ping & VMWare (perhaps not production)
- No (?) native apps have been enabled to work against an AZA/TA

Cloud Identity Summit 2013 Interop Participants

MobileIron®

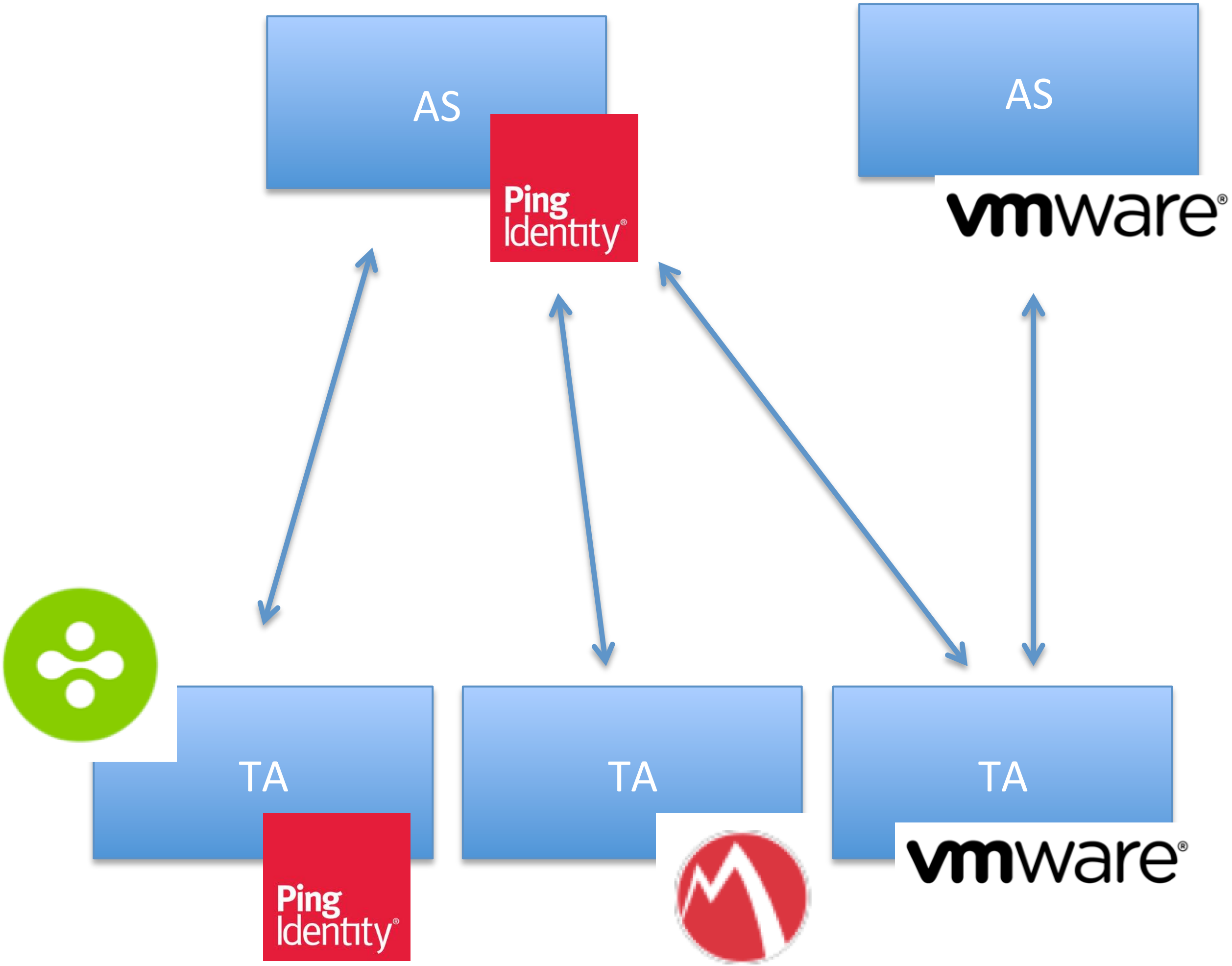
Ping Identity®

vmware®

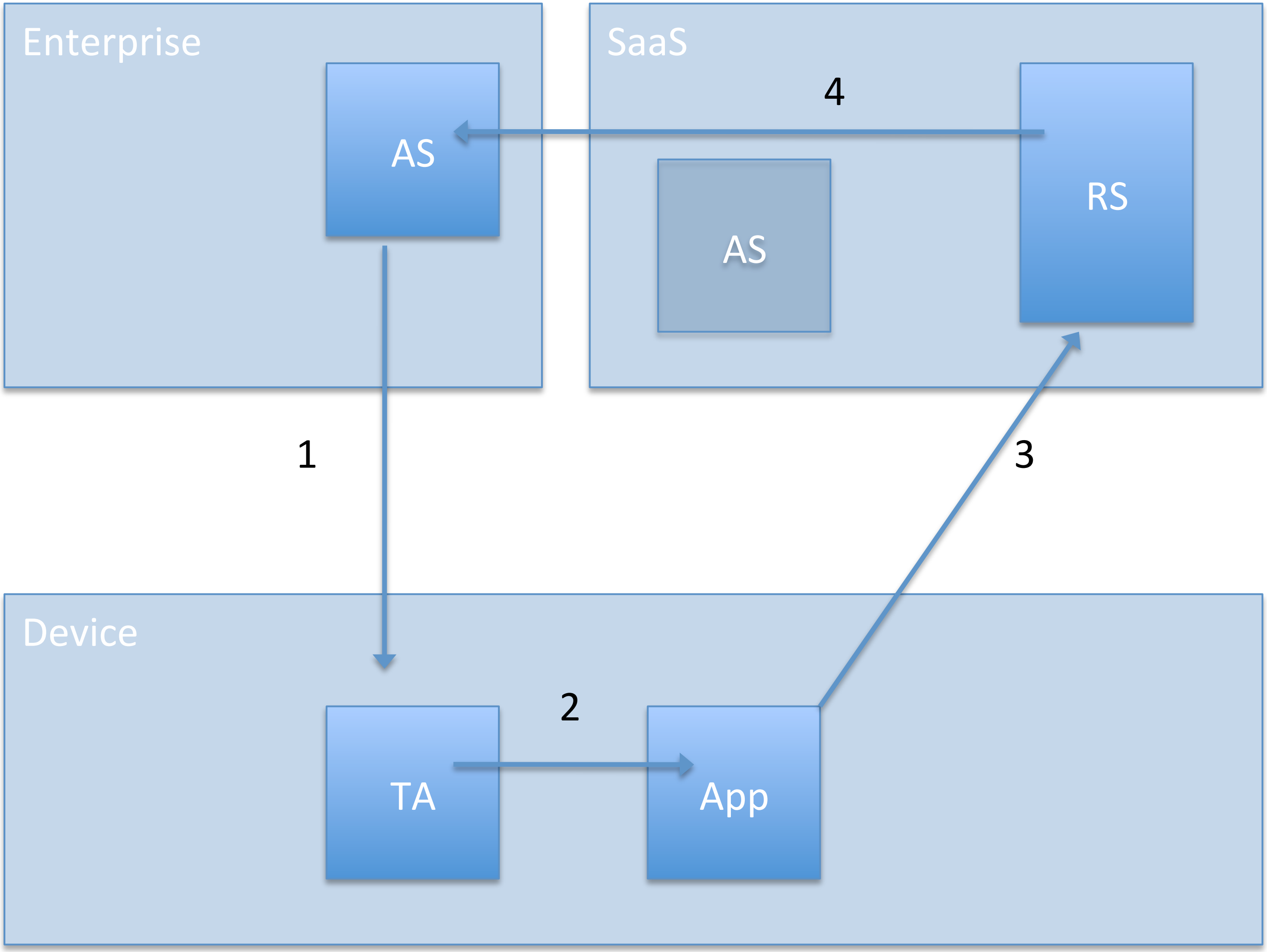
divide

box

Interop Scenarios

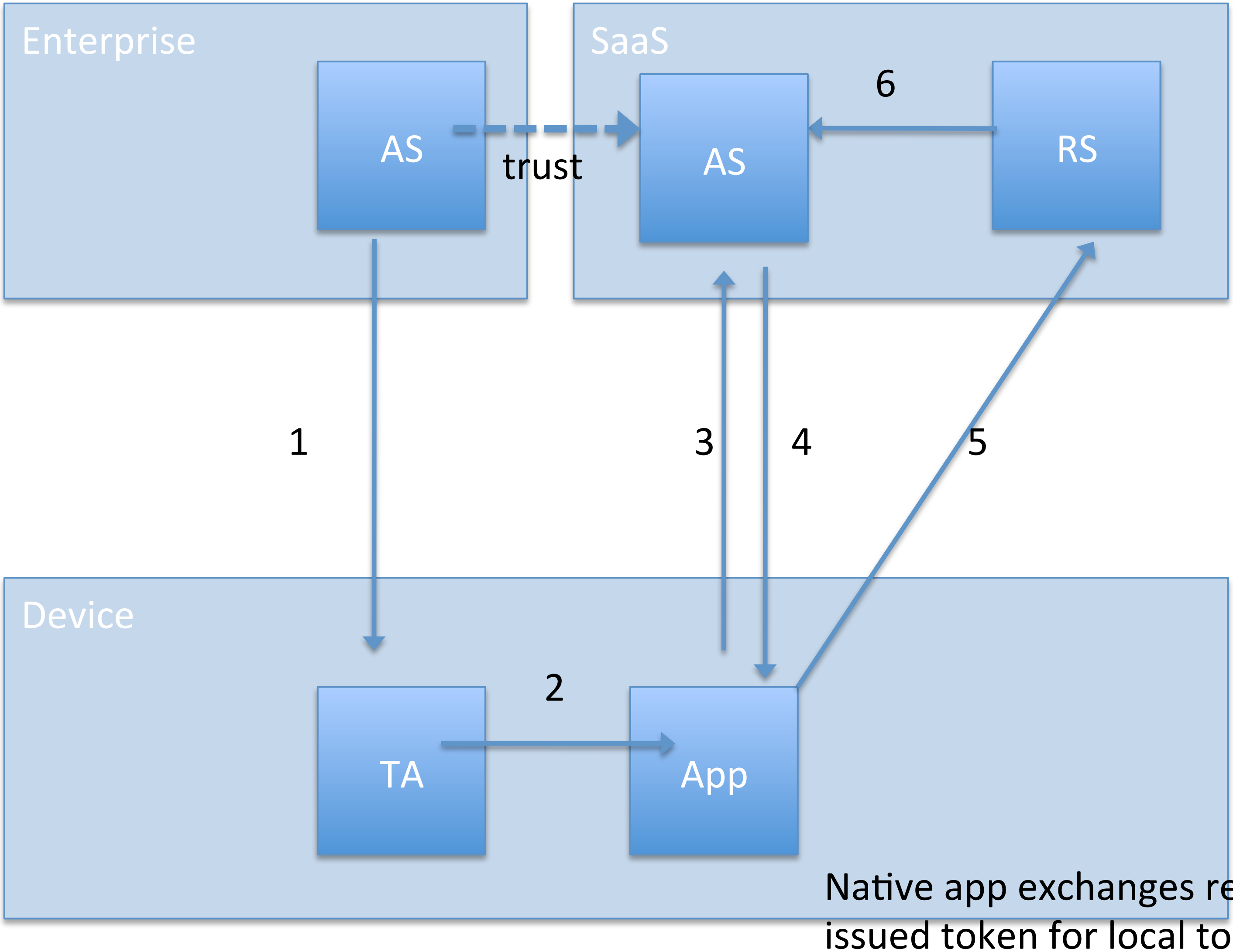


Federation – Burden on RS



RS can validate remote AS issued token, eith
by dereferencing (as shown)

Federation – Burden on native app



Native app exchanges remote AS issued token for local token