

eMONTAGE (Edge Mission-Oriented Tactical App Generator) End-User Programming on Smartphone in Tactical Environment

Objective

Enable end-users to adapt software on smartphones to meet specific mission needs.

Scenario

Imagine a U.S. soldier on patrol, deployed abroad, and walking into an unfamiliar village. Many pieces of information would be useful to that soldier in that situation. For example, it would be useful to know who the village elders are and to have pictures to identify them. It would also be useful to access information about previous spell out attacks, reports detailing the results of other contact that soldiers have had with villagers, and whether any friendly villagers speak English.

Challenges

Flexibility: Current software on smartphones is not end-user tailorable beyond personalization

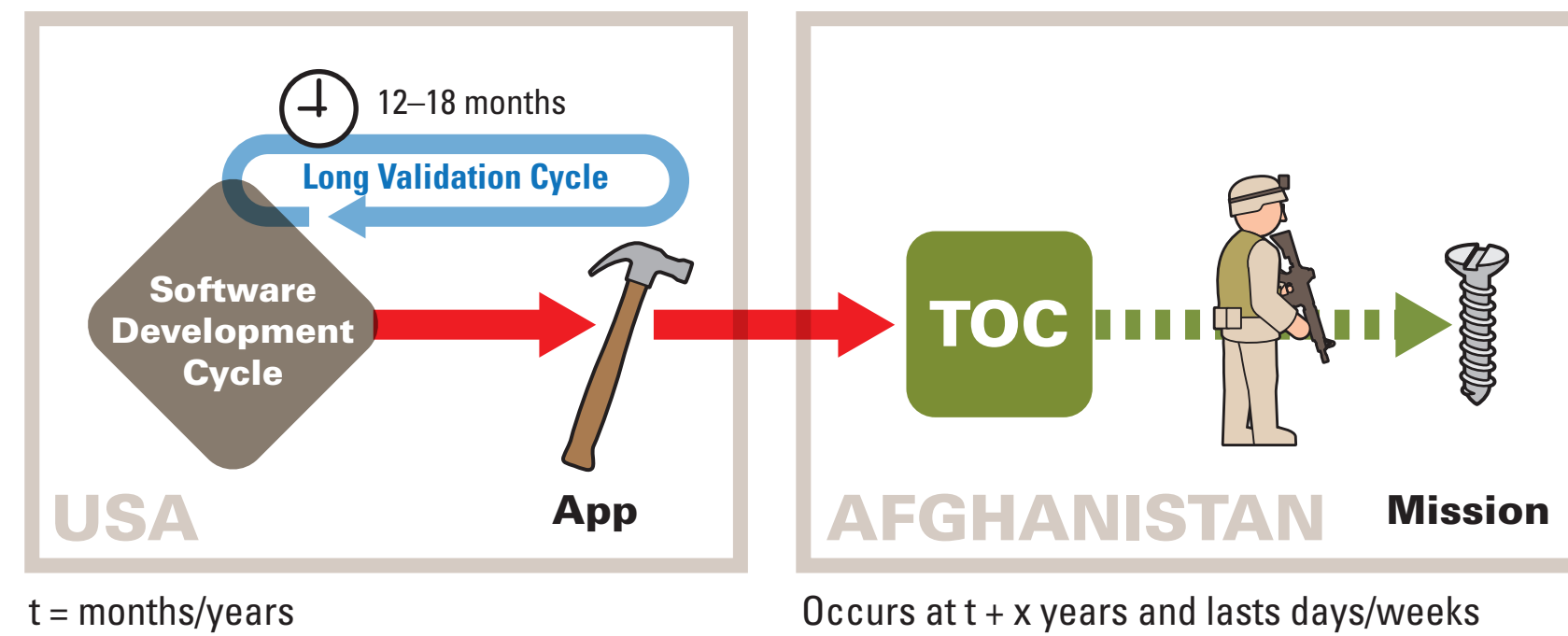
Centralization: Centralized development cannot provide quick solutions to new warfighter requirements in the field

Lack of Programming Experience: The core expertise of most end-users is not programming

Small Form Factor - A smartphone has limited display area which makes it difficult to apply traditional end-user programming paradigms (e.g., drag-and-drop)

Confidence: It takes too long to establish justified confidence in the functional and quality attributes of solutions developed in the field

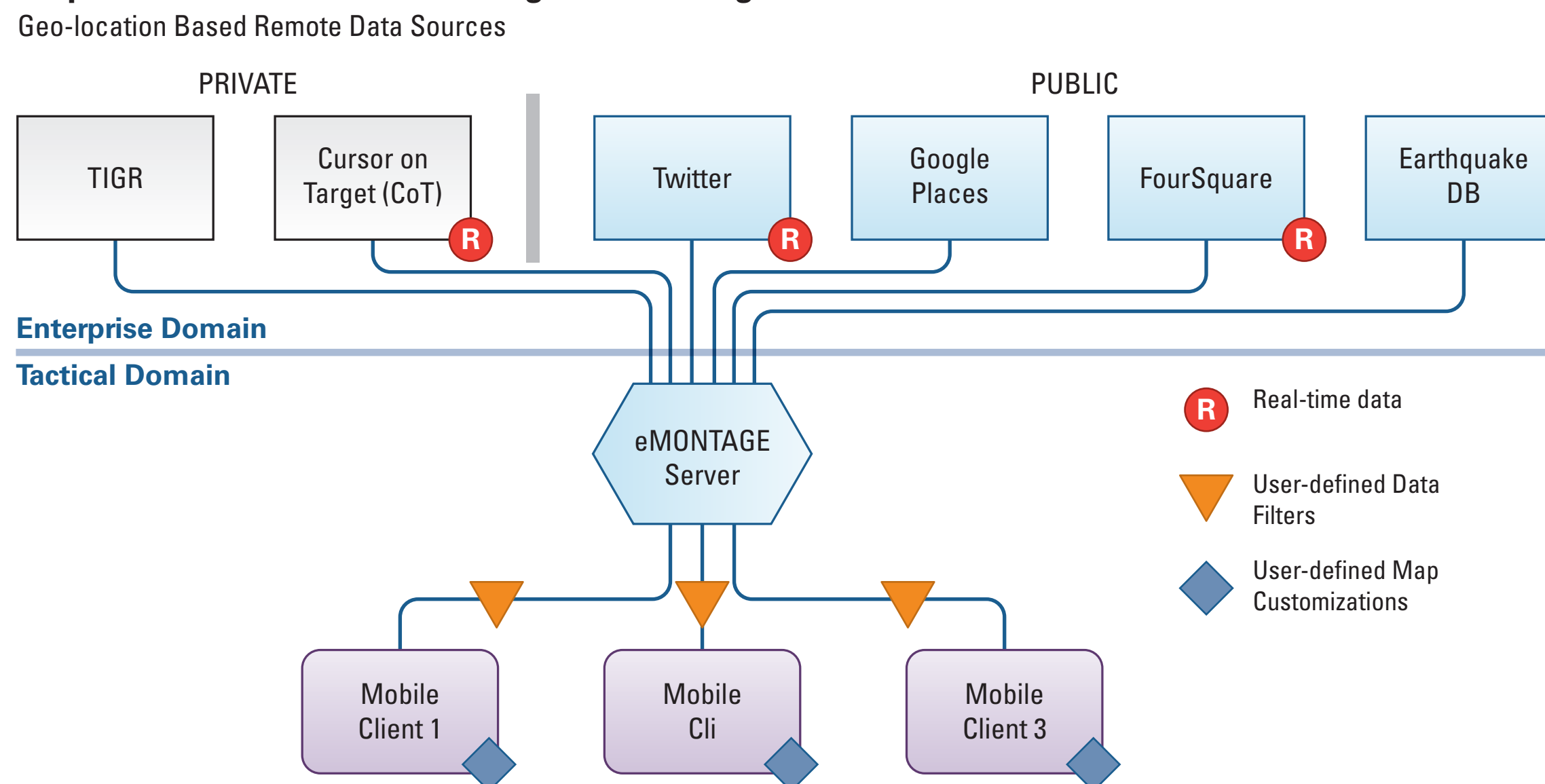
Tempo Mismatch—Development and Validation Cycle vs Mission Needs



User Interface for Dynamic List-based Forms Generated by the End-Users



Proposed Architecture for Filtering and Mashing Data from Remote Geo-Location Data Sources



Experiment Design

Provide capabilities for end-users to create a domain-specific user interface on their smartphones.

The interface must support the two most commonly used paradigms for information display on smartphones—lists and maps.

The interface must support easy input, search and retrieval, and browsing of information.

The end-user shouldn't have to write any "code"

The interface must allow the user to filter information coming from remote data sources.

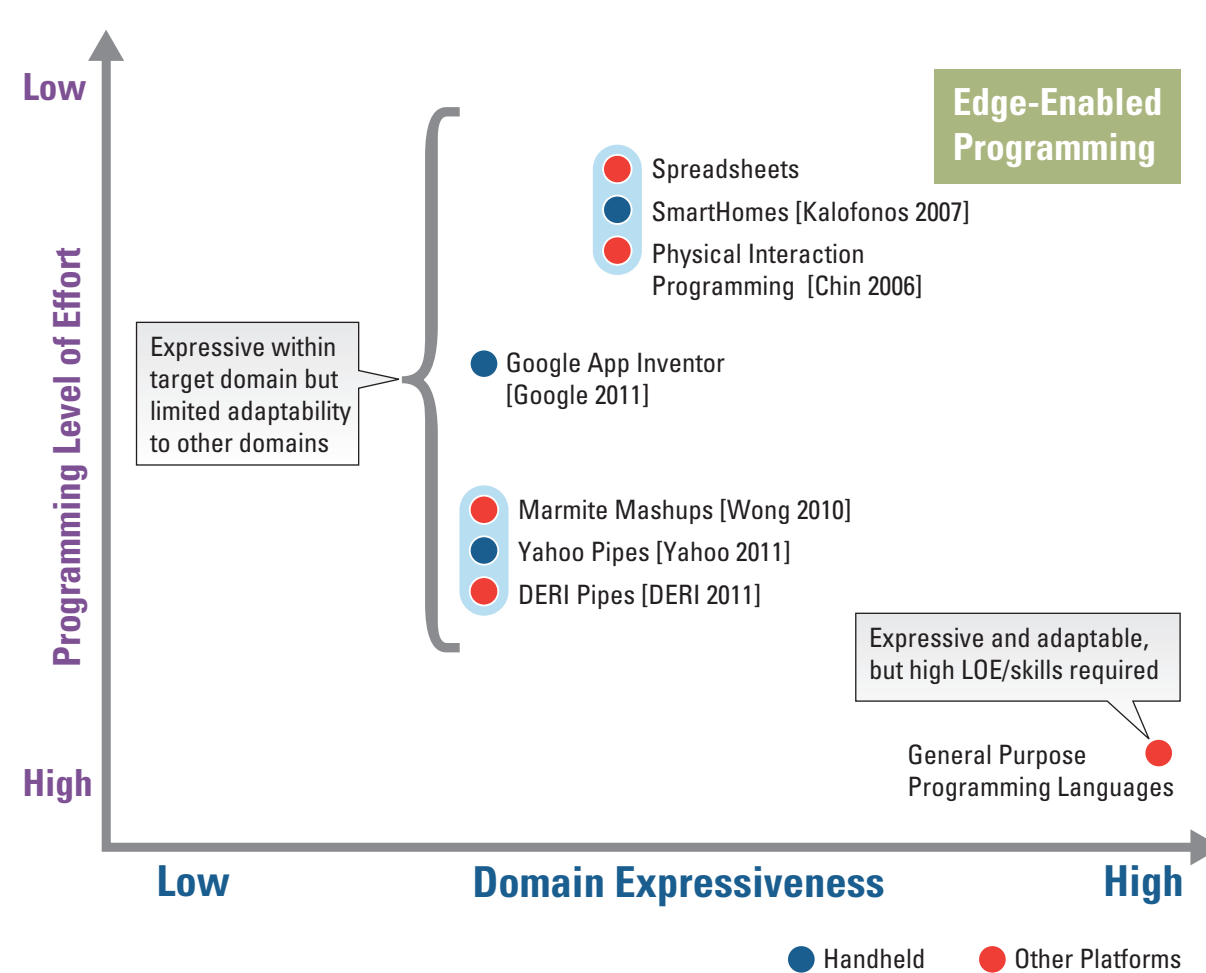
Results

Android app that enables end-users to create user-defined data types (e.g., MissingPerson, Location, Organization). The app dynamically generates list-based Android forms for each of these custom data types.

Some of our findings and observations include—

1. Younger users ("digital-natives") are more comfortable with using smartphone interfaces as compared to older users who still prefer traditional data collection approaches (e.g., paper and pencil).
2. Users do not want to be constrained by a single pre-defined data type structure. They need almost free-flow flexibility to enter and retrieve information.
3. Although other forms of inputs (e.g., natural gestures, voice recognition) are maturing on mobile devices, some cases still require text input.

Related Work



Next Steps and Considerations

Map-based Interface Allow users to customize geo-location data on a map-based interface.

Auto-detection of Data Types Support data detectors that will automatically detect common data types (dates, time, location, etc.)

Support for Real-time and Distributed Data Build mechanisms to support connection with different backend data sources.

Filtering and Mashing of Geo-location data Support content-based filtering of distributed geo-location data-based using dynamic rules defined by end-users.