



Integrating the Talkamatic Dialogue Manager with Alexa

Staffan Larsson¹, Alex Berman², Andreas Krona² and Fredrik Kronlid²

¹University of Gothenburg, Sweden

²Talkamatic AB, Sweden

sl@ling.gu.se, alex@talkamatic.se, andreas@talkamatic.se, fredrik@talkamatic.se

Abstract

This paper describes the integration of Amazon Alexa with the Talkamatic Dialogue Manager (TDM), and shows how flexible dialogue skills and rapid prototyping of dialogue apps can be brought to the Alexa platform.

1. Alexa

Amazon's Alexa ¹ is a spoken dialogue interface open to third party developers who want to develop their own Alexa "skills". Alexa has received a lot of attention and has brought renewed interest to conversational interfaces. It has strong STT (Speech To Text), TTS (Text To Speech) and NLU (Natural Language Understanding) capabilities, but provides less support in the areas of dialogue management and generation, essentially leaving these tasks to the skill developer. See Figure 1 for an overview of the Alexa architecture.

An Alexa Skill definition is more or less domain-specific. It also includes generation of natural language output, which makes it language specific. Leaving NLG to the Skill developer works fairly well when performing simple tasks but for domains demanding more complex conversational capabilities, development will be more challenging. Localizing skills to new languages will be another challenge especially if the languages is grammatically more complex than English.

2. TDM

TDM² (Talkamatic Dialogue Manager) [1, 2] is a Dialogue Manager with built-in multimodality, multilinguality, and multi-domain support, and an SDK enabling rapid development of conversational interfaces with a high degree of naturalness and usability. The basic principle behind TDM is *separation of concerns* – do not mix different kinds of knowledge. TDM keeps the following kinds of knowledge separated from each other:

- Dialogue knowledge
- Domain knowledge
- General linguistic knowledge of a particular language
- Domain-specific language
- Integration to services and data

Dialogue knowledge is encoded in the TDM DME (Dialogue Move Engine). Domain knowledge is declared in the DDD (see below). General linguistic knowledge is described in the Resource Grammar Library. Domain-specific language is described in the DDD-specific grammar. The Service and data integration is described by the Service Interface, a part of the DDD.

¹<https://developer.amazon.com/alexa>

²www.talkamatic.se

The dialogue knowledge encoded in TDM enables it to handle a host of dialogue behaviours, including but not limited to:

- Over- and other-answering (giving more or other information than requested)
- Embedded subdialogues (multiple conversational threads)
- Task recognition and clarification from incomplete user utterances
- Grounding (verification) and correction

TDM also supports localisation of applications to new languages (provided that STT and TTS is available). The currently supported and tested languages are English, Mandarin Chinese, Dutch and French. Support for more languages will be added in the future.

3. The relation Alexa – TDM

We see the combination of TDM and Alexa as a perfect match. The strengths of the Alexa dialogue platform include the nicely integrated functionality for STT, NLU, and TTS, along with the integration with the Echo hardware. The strengths of TDM are centered on the Dialogue Management component and the multilingual generation. The strengths of the two platforms are thus complementary and non-overlapping.

4. TDM Alexa integration

See Figure 2 for an overview of the Alexa-TDM integration. A wrapper around TDM receives intents (e.g. requests and questions) and slots (parameters) from Alexa, which are then translated to their TDM counterparts (request-, ask- and answer-moves) and passed to TDM. The TDM DME (Dialogue Move Engine) then handles dialogue management (updating the information state based on observed dialogue moves, and selecting the best next system move) and the utterance generation (translating the system moves into text), which are then passed back to Alexa using the TDM wrapper.

5. Dialogue Domain Descriptions

A TDM application (corresponding roughly to an Alexa skill) is defined by a DDD – a Dialogue Domain Description. The DDD is a mostly declarative description of a particular dialogue subject. Firstly, it contains information about what information (basically intentions and slots) is available in a dialogue context, and how this information is related (dialogue plans). Secondly, it contains information about how users and the system speak about this information (grammar). Lastly it contains information about how the information in the dialogue is related to the real world (service interface).

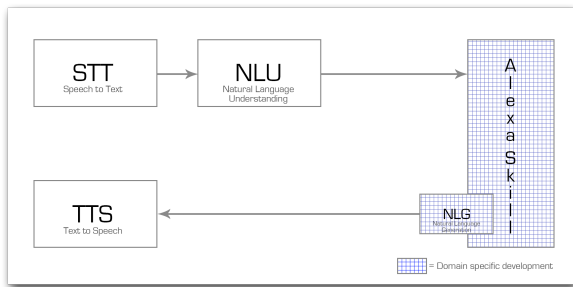


Figure 1: Alexa architecture overview.

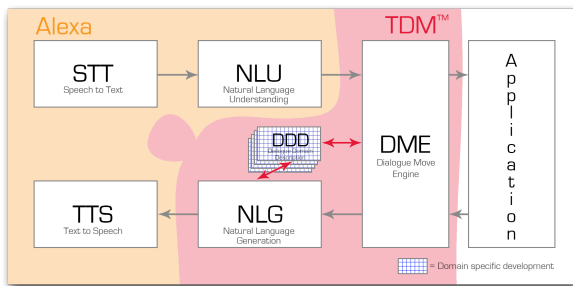


Figure 2: Alexa-TDM integration overview.

An arbitrary number of DDDs can be connected to a TDM instance for a particular user, and TDM can switch between DDDs seamlessly. This means that one can combine a number of DDDs covering different domains into one large dialogue system, where the user can talk about different subjects in the same dialogue. It can also be used as a way to allow a very modular development strategy, where different parts of a dialogue subject are developed independently. It can also be used in a freemium context, where different parts of a dialogue can be enabled or disabled depending on if the user is paying or not.

In a proof-of-concept TDM implementation of the Alexa TidePooler service³, the TDM code was approximately 25% of the code of the standard Alexa Skill. More advanced dialogue behaviors differentiate the solutions even further.

6. TDM4Music

As an demonstration of TDM-Alexa integration, we connected a TDM music app, TDM4Music, to Alexa. This app (or skill) is able to play music from Spotify, but can also answer questions. The code is mostly declarative, and uses XML for the plans and grammar descriptions. The service interface is written in Python.

6.1. Plans

The plans file contains the plan for handling all the cases when the user asks for tide information, whether none, some or all slots are filled in the user utterance. The DME logic will make sure that questions are asked to the user for getting the answers/slots missing.

³<https://www.alexaskillstore.com/Tide-Pooler/649>

6.2. Grammar

The grammar is designed to describe utterances from both user and system perspective. This makes it possible to use for generating system utterances, but also to generate a corpus of user utterances, tagged with slots and intents. The NLG facilities of TDM enable quick porting to new languages, including morphologically complex ones, without requiring grammatical knowledge on the part of the app developer.

An initial sample corpus of user utterances (needed by the Alexa NLU module) can be generated from the grammar description provided by the TDM developer as part of the DDD.

6.3. Service Interface

The service interface connects the TDM DME to the external application, in this case a Spotify service. The service interface is responsible for carrying out actions, querying services for data, checking parameter validity etc..

7. Sample dialogue

Here is a sample dialogue with the TDM4Music app, accessed through Alexa. Note that Alexa requires third party apps to be explicitly addressed, hence the "tell TDM4Music..." formulations.

- U: Alexa, tell TDM4Music to play something by Madonna
 S: What do you want to play?
 U: Which albums do you have?
 S: You can choose "Celebration", "Rebel Heart", "Like a Virgin", "Confessions..." [interrupted]
 U: Alexa!! "Like a Virgin".
 S: Playing "Like a Virgin" [plays]
 U: Alexa!
 S: [stops playing]
 U: Tell TDM4Music to play "With or without you"
 S: Playing "With or without you" [plays]
 U: Alexa!
 S: [stops playing]
 U: Ask TDM4Music which album this is?
 S: It appears on the album "The Joshua Tree"
 U: When was it released?
 S: It was released in 1987.
 U: Resume
 S: [plays]

8. Conclusion

We have shown how TDM has been integrated with Alexa to provide flexible dialogue interaction while decreasing the development effort considerably by offering built-in dialogue management and generation capabilities lacking in Alexa.

9. References

- [1] S. Larsson and A. Berman, "Domain-specific and general syntax and semantics in the Talkmatic Dialogue Manager," in *Empirical Issues in Syntax and Semantics 11*, C. Piñón, Ed. Paris: CSSP, 2016, pp. 91–110. [Online]. Available: <http://www.cssp.cnrs.fr/eiss11/>
- [2] S. Larsson, A. Berman, and J. Villing, "Adding a speech cursor to a multimodal dialogue system," in *Interspeech 2011, 12th Annual Conference of the International Speech Communication Association, Florence, Italy, 2011*, 2011, pp. 3319–3320.