This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

SMART User Manual

Program Structure

5/6/2025

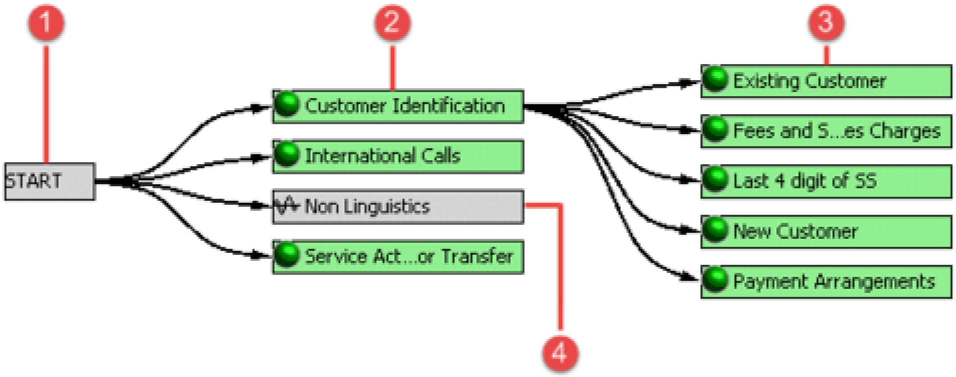
Program Structure

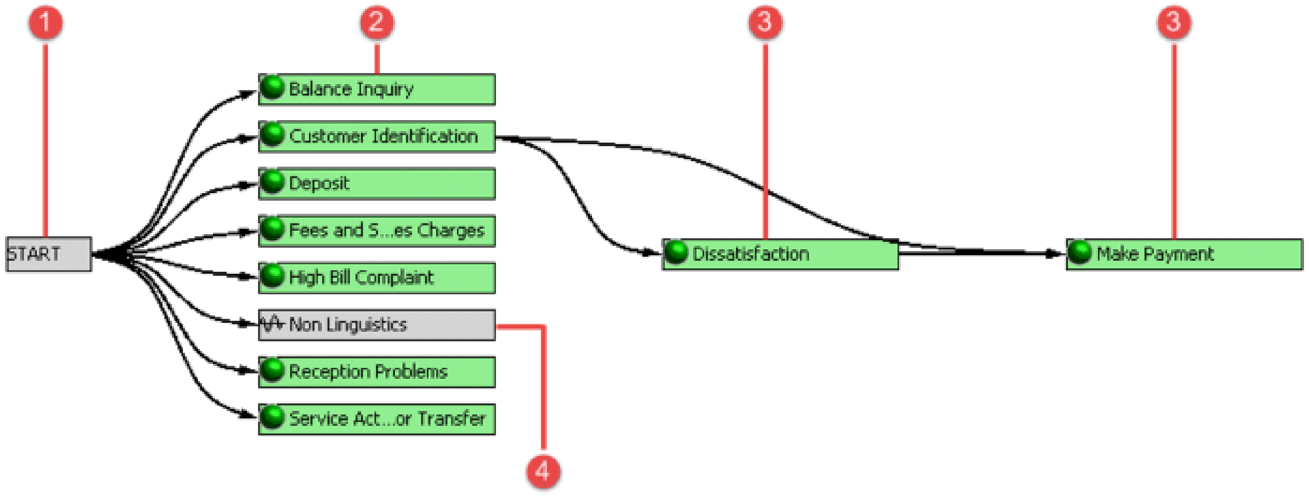
A program's structure defines the order in which SpeechMiner will search for recognition tasks (topics and non-linguistic data) in the interaction. That is, the Program's structure tells SpeechMiner what data to look for, when to look for it and where in the interaction to look for it.

Each recognition task in the program is represented by a rectangle in the program's flow chart. The order of the flowchart is determined by the arrows that connect the rectangles. An arrow is created when you create a dependent topic and it informs SpeechMiner what topic to search for and analyze next.

By default, a program's structure is defined such that all of the recognition tasks in the program are sought throughout each interaction. In the structure diagram, this is indicated by a **Start** item with separate branches to each program task.

The following images represent typical program structures:





Number	Description
1	Represents the beginning of the interaction. Start is the point from which SpeechMiner begins to review and analyze the interactions content. SpeechMiner searches for all of the topics that originate from Start.
2	Represents all of the topics that SpeechMiner will first search for and analyze in the interaction.
3	Represents dependent topics. SpeechMiner searches for a dependent Topic only after it has found the previous corresponding topic. For example: • In the image on the left, SpeechMiner will search for all 5 dependent topics only after it finds the Customer Identification topic. • In the image on the right, SpeechMiner will search for the Make Payment topic twice. Once, after it finds the Customer Identification topic, and again in interactions after it finds the Customer Identification topic and its dependent Dissatisfaction topic. The program structure is not limited to one dependency. It is possible to have numerous topic dependencies.
4	Represents non-linguistic identification. Non-linguistic identification is a part of every Program and cannot be removed.

Tip

Although you cannot create loops in a Program structure, SpeechMiner can recognize the same topic or non-linguistic data more than once in an interaction.

Related Topics

Introducing the Program Interface
Create a Program
Dependent Topic