



**JOHANNES KEPLER  
UNIVERSITY LINZ**

Bachelor's Thesis  
**Heap State Analysis View Improvement in AntTracks**

Student: FirstName LastName  
SKZ/Matr.Nr.: XXXXXXXXXX  
Email: XXX@YYY.ZZZ  
Advisor: Dipl.-Ing. Markus Weninger, BSc.  
Start date: MONTH\_ABBR 2018

---

**Dipl.-Ing. Markus Weninger**  
Institute for System Software

P +43-732-2468-4361  
F +43-732-2468-4345  
markus.weninger@jku.at

AntTracks comprises a modified Java VM based on the Hotspot VM, i.e., AntTracks VM, and an offline post-processing analysis tool, i.e., AntTracks Analyzer.

The VM's aim is to allow tracking of an application's entire life cycle by writing information about certain events to a trace file. This events include object allocations, object movements by the garbage collector, pointers between the objects and so on.

Such an event trace can then be analyzed in the offline post-processing tool. Based on the information parsed from the trace file the tool is able to reconstruct the heap for any garbage collection point.

The AntTracks Analyzer provides sophisticated functionality to classify and multi-level group objects for any heap state. Yet, the current classification system may seem hard to understand for new users.

This classification system offers functionality to create new filters (i.e., entities that exclude certain heap objects from classification) and classifiers (i.e., entities that group objects based on common properties), select filters and classifiers to apply, order the selected classifiers, and applying the selected filters and classifiers on the current heap state.

The goal of this thesis is to ease the use of the current heap state view by designing and implementing a new visualization concept.

The filter and classifier selection should try to use as much screen space as possible to reduce needed scrolling to a minimum. The user should be able to add certain classifiers to his/her favorites, preview a classifiers classification result (i.e., "How will the classification look like if I apply this classifier?") and combine multiple classifiers to reusable "Classifier chains". Features such as drag-and-drop (to select filters / classifiers) should also be included.

The above mentioned features are a minimal requirement and may be extended based on the visualization concept.

The final version of the written thesis must be submitted not later than XX.YY.ZZZZ.

**JOHANNES KEPLER  
UNIVERSITY LINZ**

Altenberger Str. 69  
4040 Linz, Austria  
www.jku.at  
DVR 0093696