Dear Readers,

We are happy to give away the fifth issue of MDT Magazine. Already five years have passed since the publishing of first MDT 1.0. This little anniversary is even more heartwarming since this year is under the sign of new Magik on Java initiative launched by our friends from GE Energy. We are very anxious to see the final release of the Smallworld. We feel it will bring us great reward as in fact MDT is the only modern IDE for Magik development neatly fitting the Java development techniques.

What’s inside? This year we bring you an overview article about Magik, Java and Smallworld. You will be able to learn the genesis of the project, its main benefits and how it will change the present way we go around Magik.

Secondly we introduce a project report nicely written by one of our project managers – Paweł Jeremicz. Paweł will share with you the details of one of the former projects. It is a good use case scenario for every modern project where you will learn about different tools and techniques used on everyday basis here in ASTEC.

The interview section brings an exclusive talk with Dirk Weidemann of house Mettenmeier. Mettenmeier is one of the latest MDT users. We have been in business relationship for many years during what time they closely observed how MDT changes and matures. Finally they have decided for a full implementation of our tool for their project efforts. The interview is their statement about the backgrounds of this decision.

We hope the Magazine will be a good lecture and we are always happy to hear your feedback.

With Best Regards,

MDT Team

Are You new to MDT?

Visit MDT online if you would like to know

- why MDT is the most intuitive IDE for Magik development,
- how it will increase your programming efficiency and
- why is it being used by companies all over the world.

Try out the 2-month trial, watch our video tutorials and subscribe to the newsletter.

Visit us today at www.mdt.net

With Java™ on board

with MDT PE 3.4 release

by Blażej Sytar, Sales & Support Leader, ASTEC

With MDT 3.4 and following 3.4.1 release we focused greatly on optimizing the performance, refactoring some old code and preparing ourselves for the coming changes announced to the whole Smallworld community – that is the introduction of Java Virtual Machine! Of course, we did not forget about adding a few nice new features.

New Extensions

- Magik on Java™ Plug-in – the first of many coming versions of a plug-in that will support the new Java VM. For starters we will provide some basic functions, like support for the new sessions. In time we will launch extended versions that will cover the major topics, as browsing and debugging. Ultimately the plug-in will be incorporated into MDT product to become the new MDT 4.0.

New Features

- Quick Session wizard – It will help you to create a whole bunch of standard sessions in just two clicks. For evaluation, demo, test purposes you can set up a standard session in no time.

- New Class Browser – the looks of the good old class browser has been enhanced to give modern feel and most of all to provide better visibility and transparency of browsed sources. The new display neatly organizes the search results, allows to unfold more detailed information and graphical representation of Magik elements provides instant feedback.

- Override indicators for methods – these are annotations that indicate methods that override methods from supertypes. After hovering mouse cursor over indicator you will see a hint with information about overridden method. From here you can easily jump to a method from supertype by just l-clicking the indicator.

Inside...

- Magik on Java™ ..............................................2-3
- Right tools make the difference! ...............4-5
- MDT in house Mettenmeier .........................6
- Get Yourself Trained! ...................................7
Magik on Java™ – one small step for GE, one giant leap for Smallworld™ community

by Bartłomiej Łączkowski, MDT Project Manager, ASTEC

Finding the motivation...

For over 20 years GE Energy has been successfully providing geospatial software solutions based on Smallworld technology to the utility and telecommunication markets as well as for public systems organizations. They have put a lot of effort into creating and maintaining a set of different products based on their technology, which consequently turned them into the global market leader for GIS. When we look inside the guts of Smallworld, two fundamental parts can be distinguished. The first one is dynamically typed & object oriented programming language called Magik. The other one is VMDS relational database that was designed from the outset to store and analyse the highly complex spatial and topological networks. Although the database has built up a great reputation for its superb topological networks. Although the database was designed from the outset to store and analyse the highly complex spatial and topological networks, it has become one of the most reliable and fastest virtual machines ever available. Although GE has invested considerable engineering talent in the support and maintenance of their own virtual machine, it seems clear that it would be great to utilise JVM and all of its goodies to run Smallworld applications, which would release a huge amount of manpower to focus on application solutions. Achieving both of these goals would not only provide a new quality for customers but also for Smallworld developers community.

The very most important question was whether it would be possible to compile Magik to Java bytecode and to run it on JVM. With the release of Java 7 that has brought the invokedynamics instruction which gave JVM a possibility to support Non-Java and dynamically typed languages, GE’s technical experts could finally meet the challenge of moving Magik towards JVM.

Building a case...

Being aware of the requirements that customers have, like i.e. having a powerful and flexible development language that has specific strengths around solving geospatial business problems or having an ability to maximize the utilization of underlying hardware infrastructure, GE found it clear that moving Magik to JVM is a heading in the right direction and it would bring a lot of benefits in the end.

First of all, utilizing JVM to run Magik will enable GE to leverage the breadth of platform and OS support that Java provides. Java is supported on all major operating systems and on number of platforms ranging from small embedded devices to large servers. However the most important aspect as GE states, is that Magik running on the JVM can result in 3–5x performance improvement in the execution of Magik code so it is almost at warp speed© Taking into account not only customer experience but also Smallworld developers community, one of the key benefits will be interoperability with the Java language and libraries. This means that it will be possible for Magik to utilize Java code and for Java code to utilize Magik. As a result the existing functionalities can leverage the enormous range of software available in the Java ecosystem. What is more, it will be possible via technology to use Java tools against Magik programs. For example, developers will have a chance to diagnose performance issues within a Magik application using Java profilers. Finally, the Magik & Java multi-language paradigm will come true. To summarize, there are a lot of benefits that one can find while thinking about Magik & Java relationship. Let’s try to point out some of the most important ones:

- Significant rise in Magik execution performance
- Support for a large set of operating systems running on anything from embedded devices, desktops, or servers
- Magik & Java interoperability, 64 bit computing, utilizing existing Java tools
- No need to rewrite existing Smallworld applications to run on JVM, the source code will just simply have to be recompiled
- MDT will become an even more “must have”

Making it real...

In 2013 GE finally announced to the community that they have started a project code-named Alchemy which, after becoming mature, was renamed as Magik on Java Technology Preview. The community members who got the access to project site can now download latest builds for evaluation purposes. After the very first look into the documentation, it can be noticed that one of GE’s key objectives is to maintain a very high degree of compatibility with existing Magik code in order to avoid a potentially large cost of porting the code to a new form. This is achieved by focusing change only to replacing existing VM and its interfaces to surrounding applications and support structures. By maintaining the behavior of the Magik runtime, the introduction of the JVM is practically transparent to existing applications.

Generally speaking, GE is augmenting Magik by using the Java Virtual Machine as a replacement for the runtime executing the Magik environment (the Magik Virtual Machine) as shown in the following diagram:

![Diagram showing the transition from Magik Virtual Machine to Java Virtual Machine](image)

**Fig 1. Technology comparison diagram**
After joining the Magik on Java Technology Preview community (you can get more info at magikonjava.feedback@ge.com) and downloading latest builds, one can i.e. play around with some demo examples of simple applications based on Magik SWIFT or Java Swing.

I can honestly say that GE made a giant leap. First of all, it can be noticed & measured that some of the Magik code can really be executed a few times faster than it was with the Magik VM. The other thing is that Magik & Java interoperability indeed works (at the moment a user can map Java static method with the use of an appropriate annotation and use it as a Magik global procedure). Summing up, although it seems like there is still a lot of work to be done by GE, they appear to be on the right track to successfully complete this project which is likely to become a next major Smallworld release.

**MDT’s new opportunities...**

Since the beginning of the Magik on Java Technology Preview project, ASTEC & GE stayed in close touch with each other. The main goal is to adapt MDT to the coming changes and to develop new features that will support upcoming Smallworld releases that will hopefully utilise Java Virtual Machine.

Recently, we published the first release of Magik on Java Plug-in that gives a user the possibility to evaluate GE’s new technology by means of interacting with new kind of Smallworld sessions in MDT. Step by step we will be developing & providing next releases with new essential features (i.e. Search & Browsing Tools, Debugger) dedicated to support Magik on Java environment.

When talking about the future of MDT, I personally believe that if Magik & Java ‘fusion’ come off, MDT together with JDT (not to mention other great plug-ins for Eclipse that support Java) will create a killer IDE with a “must have” status for every Smallworld developer. So if you are not already an MDT user, I strongly recommend to visit www.mdt.net site and download Magik Development Tools. It’s time we all became the software developers of the 21st century.

**REFERENCES :**

1. Magik on Java White Paper

2. Magik on Java Technology Preview
   http://magik-on-java-docs.s3-website-us-east-1.amazonaws.com/

**Fig 2. Magik on Java™ Plug-in - New Runtime Wizard**
Right tools make the difference!

by Pawel Jeremicz, Project Manager, ASTEC

Everyone should agree that the right choice of tools for a specific job is very important. It is no different in case of creating dedicated software. I would like to share with you a story of a project, which perfectly shows how true this statement is.

First things first. For the task management we choose Atlassian’s JIRA. You can say it has become a common standard. Together with Green Hopper and FishEye addons they provide a comprehensive support for Agile projects conducted with SCRUM methodology. We do the planning, during which we fill in the Task Backlog and distribute main responsibilities. We determine the clear objectives and discuss the concept schedule for conducting the migration. Experience carried out from similar projects helps a lot!

The entire team starts the work from configuration of the environment, in compliance with the common practices guide gathered on the project TWiki. The base for development work will be MDT Professional Edition. Since we already have a dedicated server for version control - Apache™ Subversion, the only thing left to do is to switch on the support for it in MDT. There are a few choices. Instead of Subclipse or Subversive, we go for a reliable tandem - TortoiseSVN with Context Menu plug-in, which allows for opening Windows Explorer menu directly from within Eclipse. Of course, each of us has his own little helpers, for me it is a simple plug-in, called Multi-Clipboard.

I start with the analysis of the available solutions helpful in data migrations. First thing I think of is Safe Software’s FME. Although it can handle APIC data, it would mean that our customer needs to acquire proper licenses for it. We are told that this will take a few weeks, and the clock is ticking. No other choice, but to make a manly decision – we will write a tool for importing data in ASCII format from scratch. I have an in-depth domain knowledge so I will handle the mapping between types in APIC and collections in NRM Electricity. Frequent consultations with the Product Owner become necessary to ensure customer satisfaction.

Meanwhile Krzysztof Pawlik will develop the parser and importing tool. Since MDT is based on Eclipse, extending it with the support for Python language is a matter of minutes. After installing the necessary PyDev components, Krzysiek can begin creating a tool for automatic generation of Magik classes exemplars that will be used as wrappers for parsed APIC objects. To manually create 84 classes together with countless slot access methods would be a very tedious job. Our code is swelling up very quickly. Without the use of strict Developer Style Guide it will be hard to maintain the code in good quality. KISS, DRY or YAGNI, those are only some of the rules that we go by while working on a project. A good
practice that we have followed since last year is to include MDT MStyle plug-in for QA. This Lint-like tool, preconfigured to detect a few dozen most common errors or code “bad smells”, will do us a great service.

At the same time Mariusz Dudek works on preparing engine classes responsible for processing of APIC objects, their transformation and insertion into Small-world VMDS, with due care for data integrity. This is a complex programming work with the use of many low-level operations. Errors in those methods are usually hard to detect, but you can help yourself a lot using MDT Debugger. Further elimination of mistakes is done through a thorough review. A great help during code review is Mylyn with MDT Mylyn Context Connector. It provides the context of separate tasks enhanced with the view of methods on which a specific developer was working on. The possibility to integrate Mylyn with JIRA automatizes the task lifecycle even further.

Mariusz Serbinowski bears the responsibility to verify the integrity of imported data and the quality of the complete solution. Since classes dedicated to geometric transformations are structures with high complexity of calculation logic for the validation of results, Mariusz will use the MDT MUnit support. Unit tests for methods are very helpful in discovering potential problems in early stage and test the logic independently from the UI. To minimize the time spent on bugfixing, Mariusz decides to introduce Continuous Integration. This process will be conducted by the properly configured installation of Jenkins. The Quality Assurance plan is completed by functional tests. Their preparation, execution and proper reporting is possible thanks to XStudio.

In the end the result of our work was very highly evaluated by the customer. The commitment and deep Know-How of the team is without question the most important factor in this kind of projects. Yet, without the proper choice of supporting tools we could not have done so much so quickly and in compliance with the highest quality standards.

Project in numbers:
- 408 Magik files with source code
- over 528 700 transformed APIC objects
- 27 GB of data in the result SW database
- more than 30 types of APIC objects mapped on over 50 tables in NRM Elec.
- 110 created and successfully completed tasks
- countless cups of coffee drunk 

Do you have a challenging project ahead or do you simply feel undermanned?

25 Smallworld specialists with international experience at your disposal!

E-mail: info@astec.net, Phone: +48 68 422 68 00
Błażej Sytar, ASTEC’s Sales and Support Leader talks with Dirk Weidemann, Managing Director at Mettenmeier GmbH.

[BS] Good Day Dirk. Last year you have implemented MDT for your in-house Smallworld development, but Mettenmeier has already evaluated the first version of MDT that was released over 5 years ago. What has changed that convinced you to finally give MDT a proverbial “GO”?

[DW] Good Day Błażej. That is true. We have been closely monitoring MDT development over the past years. Since the beginning we were very happy that someone took the initiative to create a modern IDE for Magik but I wouldn’t say we were fully convinced that MDT would seamlessly replace our working environments at that time. With a good dose of satisfaction we have watched MDT change and mature over the years. Finally, last year selected developers have given it a full and thorough check and the feedback was more than positive. That led us to the decision to use it here at Mettenmeier.

[BS] And how MDT now compares to MDT from a few years back?

[DW] It has changed a lot! First versions were promising and showed that you are definitely going in the right direction but we felt it isn’t enough for us. Thankfully time showed that you worked hard on improving the performance and stability of the tool as well as introducing many simplifications and improving existing solutions.

[BS] What kind of projects are you currently carrying out using MDT?

[DW] We use MDT for new projects, both internal or customer related. MDT has been also introduced to a few existing long-term projects. After some effort with adaptation and configuration it now runs very smoothly. We have observed some visible improvements in the workflow of the project compared to the previous approach.

[BS] Does MDT correspond with your needs concerning team work?

[DW] Yes, definitely. With the ability to integrate all mainstream Version Control Systems and the possibility to easily exchange session definitions it surely facilitates the work of the team, especially if this is a changing team. Another great thing is that we can benefit from the rich plug-in base provided by the Eclipse community.

[BS] Have you already observed any business benefits of the switch to MDT?

[DW] We have not made any specific ROI calculations if that is what you are asking. Nonetheless, I can honestly say that it feels that more work gets done since we use MDT. There is a visible decrease in the amount of reported bugs. Our developers also show more enthusiasm and have more fun with the development. It really helps when you do not have to pay so much attention to the language syntax or struggle with dozens of windows on your desktop – you can just spend more time on being creative and just analyze and think about what you want to write. Developers are not construction workers that need to manually lay a thousand of identical bricks one by one to build a wall. It helps a lot when the most tedious work is done automatically or semi-automatically.

[BS] Lately we hear a lot about the coming change in the Smallworld platform, which is the switch to Java Virtual Machine. Do you have any thoughts on that?

[DW] I can’t say I monitor the development of the Smallworld VM very closely but it is certainly an interesting direction. I expect a significant rise in performance. The ability to use the rich library base of Java is also very compelling.

[BS] Would you recommend MDT to other Smallworld integrators out there?

[DW] Of course. The IT business has grown and flourished over the past decade. We have seen a multitude of tools, methodologies and approaches being introduced to this business lately. It is about time that Smallworld developers have a toolkit to work with that is up to the modern standards!

[BS] Thank you for the interview.

Dirk Weidemann studied Geography and Industrial Engineering in Trier and Osnabrück (Germany). From 2002-2008 he worked as Branch Manager and Head the Utility Business Unit at SAG GmbH. He entered Mettenmeier GmbH in 2008 and has since then served as Head of IT Solutions and Managing Director of Mettenmeier’s subsidiary Conges Consulting GmbH. Since 2013 he is Managing Director of Mettenmeier GmbH.
MANY SATISFIED CUSTOMERS FROM DIFFERENT CONTINENTS ALL CLAIMING THAT THANKS TO THE TRAINING THEY ACQUIRED A COMPREHENSIVE KNOWLEDGE OF THE TOOL AND COULD UTILIZE THE MOST ADVANCED DEVELOPMENT TECHNIQUES WITHIN FEW WEEKS FROM THE INTRODUCTION!

Trainings can be held in our dedicated training room in ASTEC HQ or at your site!

2-day Standard Course

<table>
<thead>
<tr>
<th>Days</th>
<th>Topics</th>
<th>Exercises</th>
</tr>
</thead>
<tbody>
<tr>
<td>Day 1</td>
<td>1. Introduction to MDT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Installation and configuration</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. Managing Runtimes and Sessions</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4. Working with Magik projects</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>5. Magik code development techniques</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>6. Using additional tools supporting</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>7. development</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>8. Interacting with Sessions</td>
<td>✓</td>
</tr>
</tbody>
</table>

Day 2

| 9. Working with Class Browser | ✓ |
| 10. Using Magik Search tools  | ✓ |
| 11. Performing text searches  | ✓ |
| 12. Browsing Type Hierarchy   | ✓ |
| 13. Code navigation with Magik Hyperlinks | ✓ |
| 14. Using Magik Debugger     | ✓ |
| 15. Tips & Tricks            | ✓ |
| 16. Overall discussion        | ✓ |

1-day Extension Course

<table>
<thead>
<tr>
<th>Days</th>
<th>Topics</th>
<th>Exercises</th>
</tr>
</thead>
<tbody>
<tr>
<td>Day 1</td>
<td>1. Extending MDT and Eclipse</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Installing additional plug-ins</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>3. Team work in MDT</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>4. Integration with Issue Tracking System</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>5. Task-oriented development</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>6. Composing multiphase environment</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>7. Using additional useful plug-ins</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>8. Overall discussion</td>
<td>✓</td>
</tr>
</tbody>
</table>

“Even though on the first day there were many familiar issues discussed, we could learn a lot of new functions during the following training days. (...) Therefore I can recommend the training to everyone.”

Oliver Lökken, GIS Consult

Need help with environment set-up?

Except for dedicated trainings we also offer our help with more complex implementations, including adaptation and configuration of the entire developement enviroment to best fit your systems working with Smallworld™ GIS.

GET YOURSELF TRAINED!
WITH MAGIK DEVELOPMENT TOOLS

MDT Magazine
Editing & Translations: Błażej Sytar
Publisher ASTEC Sp. z o.o.
© 2014 ASTEC Sp. z o.o.
All rights reserved.

MDT is a product of
ASTEC
ADVANCED SOFTWARE TECHNOLOGIES

ul. Wyspińskiego 11
65-036 Zielona Góra, Poland
Phone: +48 68 422 68 00
Fax: +48 68 422 68 97
info@astec.net
www.astec.net
Visit our website:
www.mdt.net
www.gislet.net

Trademarks and registered trademarks. MDT Magik Development Tools is a registered trademark of ASTEC Sp. z o.o. Smallworld is a registered trademark of General Electric Company. Eclipse is a registered trademark of Eclipse Foundation, Inc. Other company or product names mentioned in this publication may be trademarks or registered trademarks of their respective companies.
Powerful solutions through advanced technologies.

As a strong GE Energy authorized Solutions Provider we offer software development services for Smallworld™ users from utilities and telecommunication industry sectors.