

Mathematica und der Finanzbereich

1. Version vom 7. Juni 2005

Zusammenfassung

Mathematica kann gewinnbringend in vielen verschiedenen Gebieten eingesetzt werden. In diesem Dokument wird knapp und prägnant auf einige Aspekte im Zusammenhang mit dem Finanzbereich eingegangen. Dabei zeigt sich die Vielseitigkeit von *Mathematica*.

Einerseits ist *Mathematica* eine Entwicklungsumgebung, in der schnell und effizient Ideen programmiert, getestet und gleichzeitig dokumentiert werden können. Andererseits kann *Mathematica* auch ausgezeichnet mit anderen Programmen kommunizieren, sei es via Clipboard, via Dateien oder via Aufruf von externem Code mit Hilfe von *MathLink*. Weiters besteht die Möglichkeit, Informationen aus dem WEB direkt in *Mathematica* einzulesen oder mittels *webMathematica* *Mathematica* Funktionen im WEB anzubieten.

Auf einige dieser Möglichkeiten wird im Folgenden etwas Näher eingegangen.

Zusätzliche Informationen und viele Links können auch auf der *Mathematica* Homepage (www.wolfram.com) und der *mec* Homepage (www.mec.li) gefunden werden.

Mathematica Funktionen und Packages

Mathematica ist gleichzeitig eine Entwicklungsumgebung, eine Dokumentationsumgebung und ein Programmierool. Die Datei [AssetPricing.pdf](http://www.mec.li/AssetPricing.pdf) (www.mec.li/AssetPricing.pdf), die als NB Datei erzeugt und als PDF Datei exportiert wurde, zeigt anschaulich, wie *Mathematica* im Finanzbereich eingesetzt werden kann. In diesem Beispiel wird auf einen kleinen Teil der *Mathematica* Funktionen im Bereich der Differentialgleichungen, der funktionalen Programmierung, der Statistik sowie der Graphiken zurückgegriffen. Auf die herausragenden Möglichkeiten im Bereich der technischen Analyse, statistischen Auswertung, Simulation, Optimierung ... wird in jenem Dokument nicht näher eingegangen.

Das Dokument *AssetPricing.pdf* verwendet ausserdem nur die fix eingebauten *Mathematica* Funktionen sowie die **Standard Packages**. (Kommerzielle) Zusatzpackages wie "**Finance Essentials**", "**Derivatives Expert**", "**UnRisk STANDARD**" und "**UnRisk PRICING ENGINE**" enthalten viele vordefinierte Funktionen für Berechnungen im Finanzbereich und nehmen einem die Programmierung vieler (bekannter) Formeln ab, wodurch sofort mit der Umsetzung eigener Ideen begonnen werden kann.

Datenaustausch via Dateien

Die einfachste Art der Kommunikation mit anderen Programmen geschieht über Datenaustausch via Dateien. *Mathematica* unterstützt rund 70 verschiedene **Dateiformate** (siehe Anhang A). Die für den Finanzbereich wichtigsten sind Text, XLS, XML. Mit Text (z.B. CSV), XLS oder XML können z.B. Excel oder OpenOffice Calc Tabellen gelesen und geschrieben werden.

Mathematica unterstützt weiters jede **Datenbank** mit einem JDBC Treiber - insbesondere MySQL, ODBC und HSQLDB. Mit einfachen Befehlen kann auf diese Datenbanken zugegriffen werden (lesen und schreiben).

Kommunikation mit externen Programmen bzw. Aufruf von externen Programmen

Mit *MathLink* können *Mathematica* Funktionen auch in externen Programmen aufgerufen werden oder es kann externer Code (DLL, Java Klassen, etc.) von *Mathematica* aus aufgerufen werden.

Ein Beispiel für den ersten Fall ist der Aufruf von *Mathematica* Funktionen von **Excel** aus mit Hilfe des kommerziellen Package "*Mathematica Link for Excel*".

Als Beispiel für den zweiten Fall kann der Aufruf einer Funktion aus der **Fincad DLL** dienen. Im Anhang B wird demonstriert wie einfach es ist, auf eine der über 1000 Fincad Funktionen von *Mathematica* aus zuzugreifen.

Die Fincad DLL ist ein kommerzielles Produkt. Alternativ könnten auch die (gratis, Open Source) **Quantlib** Funktionen (siehe: <http://wiki.quantlib.org/>) verwendet werden.

Eine interessante Möglichkeit bietet auch die direkte (von *Mathematica* aus) Manipulation der Werte und Formeln von **Calc** Tabellen. Calc ist ein (gratis) Open Source Excel Ersatz (siehe: <http://www.openoffice.org>).

Mathematica und das WEB

Mathematica bietet verschiedene Möglichkeiten des Zugangs auf WEB Ressourcen.

Mit Hilfe des Standard Package "**WebServices**" kann via SOAP und WSDL direkt auf WEB Angebote zugegriffen werden. Im Appendix C wird gezeigt, wie aktuelle Wechselkursdaten aus dem WEB abgerufen werden können.

Auch mit der Funktion "Import" kann einfach auf WEB Daten zugegriffen werden. Als Beispiel wird in Appendix D die Kursentwicklung einer Aktie heruntergeladen und geplottet.

Eine weitere Möglichkeit besteht, mittels URL Abfrage mit der Funktion "**XML`Parser`XMLGet**" auf Aktien Preise im XML Format zuzugreifen. Im Appendix E wird dies explizit gezeigt.

Mit Hilfe von **webMathematica** (*Mathematica* ist auf einem web Server installiert) und JSP (Java Server Pages) können aber auch eigene Berechnungen, die *Mathematica* Funktionen aufrufen, im Internet angeboten werden. Von einfacher Integregation über Graphik to SVG Konvertierung bis zu kompliziertesten finanzmathematischen Berechnungen ist alles denkbar (siehe z.B. <http://library.wolfram.com/explorations/webUnrisk/index.html>)

Appendices

Appendix A: Import und Export Formate

Die *Mathematica* "Import" Funktion kennt die folgenden 69 Formate

```
Length[ $\$$ ImportFormats]
 $\$$ ImportFormats

69

{"AIFF", "APS", "AU", "Bit", "BMP", "Byte", "Character16", "Character8",
 "Complex128", "Complex256", "Complex64", "CSV", "DICOM", "DIF", "Dump",
 "DXF", "EPSTIFF", "Expression", "ExpressionML", "FITS", "GIF", "HarwellBoeing",
 "HDF", "HDF5", "Integer128", "Integer16", "Integer32", "Integer64",
 "Integer8", "JPEG", "Lines", "List", "MAT", "MathML", "MGF", "MPS", "MTX",
 "NB", "NotebookML", "PBM", "PCX", "PGM", "PNG", "PNM", "PPM", "Real128",
 "Real32", "Real64", "SDTS", "SND", "STL", "String", "SymbolicXML", "Table",
 "TerminatedString", "Text", "TIFF", "TSV", "UnicodeText", "UnsignedInteger128",
 "UnsignedInteger16", "UnsignedInteger32", "UnsignedInteger64",
 "UnsignedInteger8", "WAV", "Words", "XBitmap", "XLS", "XML"}
```

"Export" kennt die gleichen Formate mit folgenden Ausnahmen: Export kennt zusätzlich

```
Complement[ $\$$ ExportFormats,  $\$$ ImportFormats]

{"AVI", "EPS", "HTML", "PDF", "PICT", "SVG", "TeX", "WMF", "XHTML+MathML"}
```

nicht jedoch

```
Complement[$ImportFormats, $ExportFormats]
{"SDTS"}
```

Appendix B: Aufruf einer Fincad Funktion

Die DLL von Fincad Developer (siehe: www.fincad.com) ermöglicht programmatischen Zugang zu über 1000 Funktionen zur finanzmathematischen Analyse, Bewertung und Messung des Risikos von Wertschriften und Derivaten. Dieser Zugang ist von *Mathematica* aus leicht zu bewerkstelligen. Im Folgenden wird gezeigt, wie die Funktion "aaWarrant" aufgerufen werden kann.

"aaWarrant" hat die folgenden Argumente:

- underlying price, exercise price, expiry date, value (settlement) date, volatility, rate - annual (Actual/365), accrual method - riskless rate (Actual/365), dividend yield, accrual method -dividend yield (actual/365), option type (call), number of shares, number of warrants, stock model type (lognormal), Statistik (alle 9), Output Variable

"aaWarrant" liefert die folgenden Werte zurück:

- fair value, delta, gamma, theta, vega, rho of rate, rho of holding cost, probability of exercise, total value of warrants

Zum Aufruf dieser Funktion in *Mathematica* müssen (einmal) die Verbindung zu NET installiert und die Funktionen geladen werden:

```
Needs["NETLink`"];
InstallNET[];
LoadNETAssembly["C:\\Programme\\fincad\\fcml8\\lib\\fcmwl.dll"];
LoadNETType["FincadFunctions"];
```

Dann kann die Funktion mit den entsprechenden Parametern bereits aufgerufen werden:

```
res = Table[1., {9}];
FincadFunctions`aaWarrant[1.275, 1.15, 37622, 37257, 0.2, 0.08, 1, 0, 1,
  1, 1000, 1000, 1, Table[i, {i, 1, 9}], res];
res
{0.11674322160499234`, 0.42076022033544486`, 0.47410776509025704`,
 -0.00013073146090393517`, 0.0015414428712496981`, 0.003886352401136109`,
 -0.005364692809276922`, 0.7883550331626363`, 116.74322160499234`}
```

Appendix C: Wechselkursdaten aus dem WEB

Dieses Beispiel zeigt, wie auf Wechselkursdaten im WEB zugegriffen werden kann.

In einem ersten Schritt werden die Packages JLink mit der Verbindung zu Java und die WebServices geladen und Java installiert.

```
Needs["JLink`"]; Needs["WebServices`"];
InstallJava[];
```

Dann wird auf den Service im WEB zugegriffen. Dazu muss natürlich eine Verbindung zum Internet vorhanden sein. Mit untenstehendem Befehl wird die verfügbare Funktion retourniert. Bei Bedarf können noch weitere Informationen zum Einsatz dieser Funktionen - insbesondere über die Anzahl und Typen der Parameter - abgerufen werden.

```
InstallService["http://www.xmethods.net/sd/2001/CurrencyExchangeService.
  wsdl"]
{getRate}
```

Wie man sieht sind dann nur einige Print Befehle notwendig, um die aktuellen Kurse für einige Währungen im Notebook einzufügen.

```
Print["Exchange Rate (für einen Dollar) am ", Drop[Date[], -3]];
Scan[Print[#, " : ", getRate["US", #]] &, {"Euro", "Switzerland", "Japan"}];

Exchange Rate (für einen Dollar) am {2005, 6, 7}

Euro: 0.8125

Switzerland: 1.244

Japan: 106.55
```

Appendix D: Herunterladen von Aktienkursen

Mit der Funktion "Import" werden die "DOW" Aktienkurse vom Januar und Februar 2005 heruntergeladen. Die URL (mit der Angabe der Firma und des Zeitraums) ist etwas kryptisch, könnte jedoch mit einer kleinen Funktion auf einfache Weise benutzerfreundlicher gestaltet werden, was hier jedoch aus Platzgründen nicht gemacht wird.

```
data = Import["http://data.moneycentral.msn.com/scripts/chrtsrv.dll?
Symbol=DOW&C2=0&C1=2&C3=0&C4=0&C5=01&C6=2005&C7=02&C8=
2005&C9=15&FileDownload=", "Table", ConversionOptions →
{"TableSeparators" → {"\r", "\n"}, {"", ""}}];
```

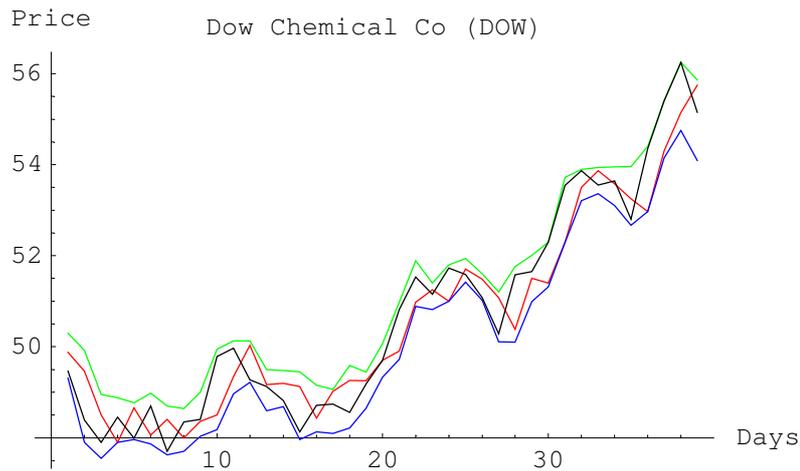
Im Folgenden werden die ersten 10 Zeilen dargestellt:

```
data[[Range[10], All]] // ColumnForm

{"\US and Canadian historical stock prices and US indices prices provided by CSI",
"\US historical mutual fund and industry prices provided by Media General Financ.
"\International stock prices and International indices prices provided by Internat
"Dow Chemical Co (DOW)"
"Daily prices"
"DATE", "OPEN", "HIGH", "LOW", "CLOSE", "VOLUME"
{{2005, 2, 28}, 55.75`, 55.86`, 54.09`, 55.15`, 6326200}
{{2005, 2, 25}, 55.14`, 56.25`, 54.75`, 56.24`, 4604200}
{{2005, 2, 24}, 54.3`, 55.4`, 54.15`, 55.4`, 6706800}
{{2005, 2, 23}, 52.97`, 54.4`, 52.97`, 54.35`, 4695400}
```

Aus dieser Liste werden die Tagespreise extrahiert und geplottet, wobei die verschiedenen Farben Open (rot), High (grün), Low (blau), Close (schwarz) Preise repräsentieren.

```
coredata = Part[data, Range[7, Length[data]]];
Show[
ListPlot[Reverse[Transpose[coredata][[#[[1]]]], PlotJoined → True,
PlotStyle → #[[2]], DisplayFunction → Identity] & /@
{{2, Red}, {3, Green}, {4, Blue}, {5, Black}},
PlotLabel → StringDrop[StringDrop[ToString[data[[4]]], 1], -1], Ticks →
{Automatic, Automatic}, AxesLabel → {"Days", "Price"}, ImageSize →
{300, 200},
DisplayFunction → $DisplayFunction];
```



Appendix D: Herunterladen von Aktienkursen im XML Format

Dieses Beispiel benutzt XMLToday, um mittels URL Abfrage Aktienpreise im XML Format zu erhalten. Die verlangten Aktien werden hinter dem "?s=" Parameter spezifiziert.

```
xmlquotes = XML`Parser`XMLGet["http://www.xmltoday.com/examples/stockquote/
getxmlquote.vep?s=ba+lmt+ibm+int+amd+mmm"] ;
```

Im nächsten Schritt wird eine Funktion definiert, die die gewünschte Information extrahiert und in eine Form umwandelt, die als Input zum Plotten verwendet werden kann.

```
GetQuotes[xml_] := Cases[xml, XMLElement["stock_quote", _, {___, XMLElement[
"symbol", _, {symbol_}], ___, XMLElement["price", {"type" -> "ask",
"value" -> v_}, {}], ___]} -> {symbol, ToExpression[v]}, Infinity];
```

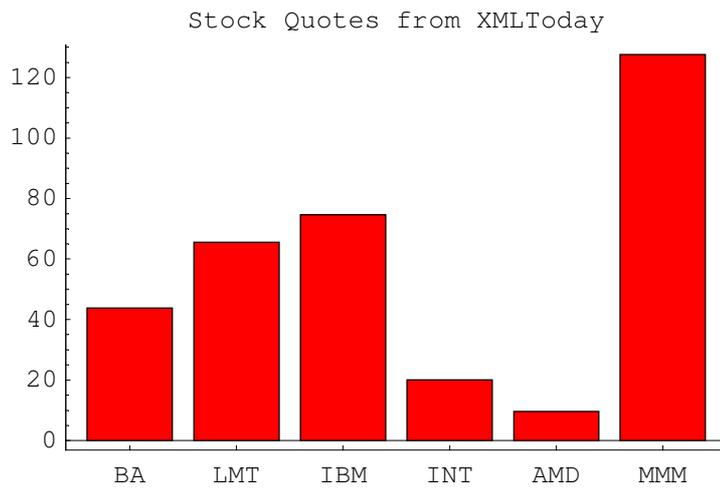
Auf die eingelesenen Daten angewendet resultiert:

```
quotes = GetQuotes[xmlquotes]

{{"BA", 43.83`}, {"LMT", 65.51`}, {"IBM", 74.65`}, {"INT", 20.04`}, {"AMD",
9.67`}, {"MMM", 127.55`}}
```

und geplottet:

```
Needs["Graphics`Graphics`"];
BarChart[#[[2]] & /@quotes, BarLabels -> (#[[1]] & /@quotes), Frame -> {True,
True, False, False}, ImageSize -> 250, PlotLabel -> "Stock Quotes from
XMLToday"];
```



Beachte: die URL von XMLToday ist (derzeit) nicht mehr aktiv.