



## Kapitel 1

# Entwickeln Sie Ihre Handelssysteme noch von Hand?

Die Entwicklung erfolgreicher Handelssysteme gehört seit jeher zu den größten Herausforderungen im quantitativen Trading. Obwohl heute leistungsfähige Computer, umfangreiche historische Marktdaten und eine nahezu unbegrenzte Rechenleistung zur Verfügung stehen, entstehen viele Handelsstrategien noch immer nach demselben Schema wie vor zwanzig Jahren.

- Eine Idee wird formuliert.
- Die Handelsregeln werden programmiert.
- Anschließend beginnt die Optimierung der Parameter.
- Danach folgt der Backtest.

Fallen die Ergebnisse zufriedenstellend aus, wird die Strategie im Paper Trading oder direkt im Live-Handel eingesetzt.

Dieses Vorgehen funktioniert – zumindest gelegentlich. Gleichzeitig besitzt es jedoch einen entscheidenden Nachteil: Der Entwickler sucht selbst nach einer Handelsidee.

Damit ist bereits zu Beginn festgelegt, in welche Richtung die Suche überhaupt verläuft. Gute Ideen entstehen häufig durch Erfahrung, Intuition oder Zufall. Ebenso viele interessante Zusammenhänge bleiben jedoch unentdeckt, weil niemand gezielt nach ihnen gesucht hat.

Hinzu kommt ein weiteres Problem, das jedem Entwickler systematischer Handelsstrategien bekannt ist.

## Die Optimierungsfalle

Fast jede Strategie lässt sich durch geschickte Wahl der Parameter verbessern.

Ein gleitender Durchschnitt von 21 Tagen liefert vielleicht eine ansprechende Performance.

Mit 24 Tagen sieht das Ergebnis noch etwas besser aus.

Bei 27 Tagen steigt der Gewinn erneut.

Spätestens jetzt beginnt jedoch die eigentliche Gefahr.

Die Strategie wird immer besser an die Vergangenheit angepasst – ohne dass sie dadurch zwangsläufig robuster für die Zukunft wird.

Dieses Phänomen wird als **Overfitting** oder **Curve Fitting** bezeichnet.

Die Strategie lernt nicht die Gesetzmäßigkeiten des Marktes. Sie lernt lediglich die Besonderheiten der historischen Daten kennen.

Im Live-Handel zeigt sich dann häufig ein völlig anderes Bild. Die Performance bleibt hinter den Erwartungen zurück, Drawdowns fallen höher aus und viele vermeintlich hervorragende Strategien verlieren ihren statistischen Vorteil bereits nach kurzer Zeit.

## Ein anderer Ansatz

Genau an diesem Punkt setzt **Genetic Programming** an.

Anstatt eine einzelne Idee zu entwickeln und anschließend zu optimieren, übernimmt der Computer einen großen Teil der eigentlichen Forschungsarbeit.

Aus tausenden möglicher Regelbestandteilen, Filter und Einstiegssignalen entstehen automatisch vollständige Handelssysteme.

Diese werden anschließend anhand objektiver Kriterien bewertet, miteinander verglichen und durch zahlreiche Robustheitstests geprüft.

Nicht der Entwickler entscheidet also, welche Handelsidee untersucht wird. Vielmehr durchsucht die Software einen riesigen Lösungsraum nach Regelkombinationen, die statistisch überzeugende Eigenschaften besitzen.

Dadurch entstehen häufig Handelsmodelle, auf die ein Mensch allein kaum gekommen wäre.

| INFOBOX                                                                                          |                            |
|--------------------------------------------------------------------------------------------------|----------------------------|
| So verändert Genetic Programming den Entwicklungsprozess                                         |                            |
| Klassisch                                                                                        | Genetic Programming        |
| Idee entwickeln                                                                                  | Suchraum definieren        |
| Programmieren                                                                                    | Strategien erzeugen        |
| Optimieren                                                                                       | Tausende Varianten testen  |
| Backtest                                                                                         | Robustheit prüfen          |
| Bewertung                                                                                        | Beste Strategien auswählen |
| Genetic Programming ersetzt den Entwickler nicht.<br>Es erweitert seine Möglichkeiten erheblich. |                            |

## Von der Idee zum Research-Prozess

Moderne Genetic-Programming-Systeme verändern deshalb nicht nur die Geschwindigkeit der Strategieentwicklung, sondern den gesamten Forschungsprozess.

Statt Wochen oder Monate in die Entwicklung einzelner Strategien zu investieren, können innerhalb kurzer Zeit (oft wenige Stunden!) Hunderte oder sogar Tausende unterschiedlicher Ansätze erzeugt und nach objektiven Kriterien bewertet werden.

Der Entwickler übernimmt dabei eine neue Rolle. Er programmiert nicht mehr jede Handelsidee selbst. Er definiert vielmehr die Rahmenbedingungen des Research-Prozesses, bewertet die Ergebnisse und entscheidet anschließend, welche Strategien den hohen Anforderungen eines professionellen Einsatzes genügen.

## Warum diese Broschüre?

In den vergangenen Jahren konnten wir mehrere professionelle Genetic-Programming-Plattformen intensiv in der Praxis einsetzen und miteinander vergleichen.

Zu den bekanntesten Vertretern gehören heute:

- **Build Alpha**
- **GSB (Genetic System Builder)**
- **TSL Pro (Trading System Lab Professional)**

Alle drei Programme verfolgen unterschiedliche Ansätze und gehören zweifellos zur Spitzengruppe professioneller Werkzeuge für die automatisierte Entwicklung regelbasierter Handelsstrategien.

Diese Broschüre soll keine wissenschaftliche Abhandlung sein !

Sie richtet sich an Trader, Vermögensverwalter und quantitative Research-Abteilungen, die einen praxisnahen Überblick über Genetic Programming erhalten möchten und wissen möchten, warum wir uns nach intensiver praktischer Nutzung für den Vertrieb von Build Alpha im deutschsprachigen Raum entschieden haben.





## Kapitel 2

# Genetic Programming – wenn der Computer selbst nach Handelsstrategien sucht

Die meisten Entwickler automatischer Handelssysteme beginnen ihre Arbeit mit einer konkreten Idee.

Beispielsweise:

- Kaufe, wenn der 20-Tage-Durchschnitt den 50-Tage-Durchschnitt von unten nach oben kreuzt.
- Verkaufe, wenn der RSI einen bestimmten Schwellenwert überschreitet.
- Nutze einen ATR-basierten Stop-Loss.
- Verwende einen Volatilitätsfilter.

Alle diese Ideen stammen ursprünglich vom Entwickler. Der Computer übernimmt anschließend lediglich die Berechnung und Optimierung.

Genetic Programming verfolgt einen grundsätzlich anderen Ansatz. Hier entwickelt nicht mehr der Mensch die Handelsregeln. Der Computer erzeugt sie selbst.

## Von der Idee zur automatischen Strategiegenerierung

Genetic Programming gehört zu den Verfahren der Künstlichen Intelligenz und orientiert sich an den Prinzipien der biologischen Evolution.

Dabei werden nicht einzelne Parameter optimiert, sondern vollständige Handelssysteme erzeugt.

Jede Strategie besteht aus einer Kombination verschiedener Elemente, beispielsweise:

- Einstiegssignale
- Ausstiegsregeln
- Filter
- Zeitbedingungen
- Volatilitätskriterien
- Stop-Loss-Regeln
- Gewinnziele
- Positionsmanagement

Diese Kombination bildet ein vollständiges Regelwerk.

Der Computer erzeugt zunächst eine große Anzahl unterschiedlicher Strategien und bewertet jede einzelne anhand zuvor definierter Qualitätskriterien.

Zu diesen Kriterien (**Lernzielen**) gehören beispielsweise:

- Nettogewinn
- Profit Factor
- Sharpe Ratio
- Drawdown
- Robustheit
- Stabilität
- Anzahl der Trades
- Risiko-Rendite-Verhältnis
- **Oder auch (besonders interessant): der Abstand der Performance einer Strategie zu der einer theoretisch optimalen Strategie (und diese lässt sich ja historisch betrachtet generieren)**

Die erfolgreichsten Strategien werden anschließend weiterentwickelt. Weniger erfolgreiche Varianten werden verworfen. Dieser Prozess wiederholt sich über viele Generationen.

Auf diese Weise entstehen nach und nach Handelsmodelle, die statistisch deutlich überzeugender sind als ihre ursprünglichen Ausgangsversionen.

**Wichtiger Aspekt hierbei ist es, die Komplexität der Regelwerke so stark wie möglich zu begrenzen um Overfitting weitestgehend vermeiden zu können. Dies lässt sich in Build Alpha leicht einstellen auf z.B. max 2 Einstiegsregeln und max eine Ausstiegsregel. Weiterhin kann man in Build Alpha die Logiken gleichzeitig auf mehreren Symbolen entwickeln lassen um beispielsweise nach Logiken zu suchen, die nicht nur den S&P 500 handeln können sondern auch NASDAQ, Dow Jones und Russel !**

## Evolution statt Vermutung

Der wesentliche Unterschied liegt darin, dass der Entwickler keine fertige Handelsidee mehr formulieren muss. Stattdessen definiert er den Suchraum.

Er legt beispielsweise fest,

- welche Märkte untersucht werden,
- welche Indikatoren verwendet werden dürfen,
- welche Handelszeiten zulässig sind,
- welche Risikogrenzen gelten,
- und nach welchen Kriterien Strategien bewertet werden sollen.

Die eigentliche Suche übernimmt anschließend die Software.

Dadurch werden häufig Zusammenhänge entdeckt, die einem menschlichen Entwickler niemals aufgefallen wären.

Gerade diese Fähigkeit macht Genetic Programming zu einem der interessantesten Werkzeuge der modernen quantitativen Forschung.

## **Geschwindigkeit verändert den gesamten Forschungsprozess**

Ein weiterer entscheidender Vorteil liegt in der enormen Geschwindigkeit.

Während ein Entwickler manuell vielleicht einige wenige Handelsideen pro Woche untersuchen kann, analysieren moderne Genetic-Programming-Systeme innerhalb weniger Stunden Tausende möglicher Regelkombinationen.

Dadurch verschiebt sich der Schwerpunkt der Forschungsarbeit. Nicht mehr das Programmieren einzelner Strategien steht im Mittelpunkt.

Viel wichtiger wird die objektive Bewertung der Ergebnisse.

Professionelle Anwender investieren deshalb einen großen Teil ihrer Zeit nicht in das Schreiben von Handelsregeln, sondern in die Auswahl robuster Strategien sowie deren statistische Validierung.

## **Robustheit ist wichtiger als der höchste Gewinn**

Eine Strategie mit dem höchsten Gewinn im Backtest ist nicht zwangsläufig die beste Strategie. Im Gegenteil.

Besonders hohe historische Gewinne sind häufig ein Hinweis auf eine zu starke Anpassung an die Vergangenheit. Professionelle Entwickler achten deshalb wesentlich stärker auf die Robustheit eines Handelssystems.

Dazu gehören unter anderem:

- Walk-Forward-Analysen
- Monte-Carlo-Simulationen
- Noise-Tests
- Out-of-Sample-Tests
- Parameterstabilität
- Forward-Analysen

Nur Strategien, die auch diese Prüfungen bestehen, kommen für einen praktischen Einsatz überhaupt in Betracht.



|                                                                                |
|--------------------------------------------------------------------------------|
| INFOBOX 02                                                                     |
| Die drei Säulen robuster Handelssysteme                                        |
| 1. Gute Handelsidee                                                            |
| ✓ Logisch nachvollziehbar                                                      |
| ✓ Ausreichende Anzahl Trades                                                   |
| 2. Statistische Validierung                                                    |
| ✓ Walk Forward                                                                 |
| ✓ Monte Carlo                                                                  |
| ✓ Out-of-Sample                                                                |
| ✓ Noise Tests                                                                  |
| 3. Portfoliointegration                                                        |
| ✓ Geringe Korrelation                                                          |
| ✓ Sinnvolle Diversifikation                                                    |
| ✓ Angemessenes Risiko                                                          |
| Robuste Strategien entstehen erst durch das Zusammenspiel aller drei Bereiche. |

## Genetic Programming ersetzt den Entwickler nicht

Ein häufiges Missverständnis besteht darin, dass Genetic Programming den Entwickler überflüssig macht.

Das Gegenteil ist der Fall.

Die Qualität der Ergebnisse hängt entscheidend davon ab,

- welche Märkte untersucht werden,
- welche Eingangsdaten verwendet werden,
- welche Fitness-Kriterien gewählt werden,
- welche Robustheitstests durchgeführt werden,
- und wie die gefundenen Strategien später bewertet werden.

Die Software übernimmt die Suche. Die eigentliche Forschungsarbeit bleibt jedoch Aufgabe des Entwicklers.

Gerade diese Kombination aus menschlicher Erfahrung und automatisierter Strategiegenerierung führt häufig zu deutlich besseren Ergebnissen als einer der beiden Ansätze allein.

## Von der Theorie zur Praxis

In den vergangenen Jahren haben sich mehrere spezialisierte Softwarelösungen etabliert, die Genetic Programming für die Entwicklung automatischer Handelssysteme einsetzen.

Zu den bekanntesten Vertretern gehören heute:

- **Build Alpha**
- **GSB (Genetic System Builder)**
- **TSL Pro (Trading System Lab Professional)**

Alle drei Programme verfolgen dasselbe Grundprinzip, unterscheiden sich jedoch deutlich hinsichtlich Bedienung, Workflow, Funktionsumfang und Zielgruppe.

Im nächsten Kapitel stellen wir diese drei Plattformen zunächst kurz vor, bevor wir anschließend ausführlich auf Build Alpha eingehen und unsere praktischen Erfahrungen aus vielen Monaten intensiver Nutzung beschreiben.

---

### **Research-Prozess mit Genetic Programming**

1. Markt auswählen
2. Suchraum definieren
3. Strategien automatisch erzeugen
4. Backtests durchführen
5. Robustheit prüfen
6. Beste Strategien auswählen
7. Code exportieren
8. Live validieren

## **Kapitel 3**

# **Professionelle Genetic-Programming-Systeme im Vergleich**

Noch vor wenigen Jahren war Genetic Programming vor allem Universitäten, Forschungseinrichtungen und einigen wenigen Hedgefonds vorbehalten. Die Entwicklung entsprechender Software war aufwendig, erforderte umfangreiche Programmierkenntnisse und setzte häufig eigene Entwicklungsabteilungen voraus.

In den vergangenen Jahren hat sich dieses Bild grundlegend verändert.

Heute stehen mehrere kommerzielle Plattformen zur Verfügung, mit denen auch kleinere Research-Teams, Vermögensverwalter oder ambitionierte Privatanleger professionelle Genetic-Programming-Verfahren einsetzen können.



Zu den bekanntesten Lösungen gehören heute:

- **Build Alpha**
- **GSB (Genetic System Builder)**
- **TSL Pro (Trading System Lab Professional)**

Alle drei Programme verfolgen dasselbe Ziel:

**Die automatisierte Entwicklung robuster, regelbasierter Handelsstrategien.**

Die Wege dorthin unterscheiden sich jedoch teilweise deutlich.

---

## GSB – Genetic System Builder

GSB gehört seit vielen Jahren zu den etablierten Programmen im Bereich der automatisierten Strategieentwicklung.

Die Software bietet einen sehr großen Funktionsumfang und richtet sich vor allem an erfahrene Systementwickler, die ihre Strategien bis ins Detail konfigurieren möchten.

Zu den besonderen Stärken zählen unter anderem:

- Unterstützung mehrerer Datenserien innerhalb einer Strategie
- umfangreiche Walk-Forward-Analysen
- leistungsfähige Optimierungsverfahren
- Cloud-Unterstützung für verteilte Berechnungen
- sehr flexible Konfigurationsmöglichkeiten

Gerade diese große Flexibilität führt allerdings auch dazu, dass der Einstieg für neue Anwender vergleichsweise anspruchsvoll ist.

Wer bereit ist, sich intensiv mit der Software auseinanderzusetzen, erhält ein äußerst leistungsfähiges Werkzeug für professionelle Forschungsprojekte.

---

## TSL Pro – Trading System Lab Professional

Auch TSL Pro zählt seit vielen Jahren zu den bekanntesten kommerziellen Lösungen im Bereich Genetic Programming.

Die Software legt besonderen Wert auf die automatische Entwicklung vollständiger Handelssysteme und unterstützt unter anderem:

- Long- und Short-Strategien
- Portfolioansätze
- Pair-Trading
- umfangreiche Out-of-Sample-Tests
- automatischen Codeexport für verschiedene Handelsplattformen

TSL Pro richtet sich vor allem an professionelle Anwender, die einen sehr großen Funktionsumfang benötigen und bereit sind, sich intensiv in die Arbeitsweise der Software einzuarbeiten.

Der Einstieg ist entsprechend anspruchsvoll, die Möglichkeiten sind dafür außerordentlich umfangreich.

---

## Build Alpha

Build Alpha verfolgt einen etwas anderen Ansatz.

Die Software richtet sich ebenfalls an professionelle Anwender, legt jedoch besonderen Wert auf einen möglichst effizienten Entwicklungsprozess.

Viele Arbeitsschritte, die bei anderen Programmen umfangreiche Konfigurationen erfordern, lassen sich innerhalb weniger Minuten durchführen.

Dadurch eignet sich Build Alpha sowohl für institutionelle Research-Abteilungen als auch für erfahrene Privatanleger, die ihre Zeit lieber in die Bewertung von Strategien als in die Konfiguration der Software investieren möchten.

Nach unserer Erfahrung ist dies einer der größten Unterschiede zu vielen anderen Programmen dieser Klasse.

---

## Unsere Erfahrungen

In den vergangenen Monaten konnten wir alle drei Programme intensiv im praktischen Einsatz vergleichen.

Unser Ziel bestand dabei nicht darin, den Funktionsumfang einzelner Programme miteinander zu vergleichen, sondern die Frage zu beantworten:

**Mit welcher Plattform lässt sich ein professioneller Research-Prozess am effizientesten durchführen?**

Dabei standen insbesondere folgende Kriterien im Mittelpunkt:

- Bedienbarkeit
- Geschwindigkeit der Strategieentwicklung
- Qualität der Robustheitstests
- Codeexport
- Stabilität des Workflows
- Aufwand bis zur fertigen Strategie
- Preis-Leistungs-Verhältnis

Dabei zeigte sich sehr schnell, dass alle drei Programme professionelle Ergebnisse liefern können.

Die Unterschiede liegen weniger in der Qualität der erzeugten Strategien als vielmehr im Arbeitsablauf und in der Benutzerführung.

Gerade hier konnte Build Alpha aus unserer Sicht besonders überzeugen.

---

## Vergleich auf einen Blick

| Merkmal                    | Build Alpha            | GSB                 | TSL Pro             |
|----------------------------|------------------------|---------------------|---------------------|
| Bedienung                  | einfach                | eher aufwändig      | eher aufwändig      |
| Einarbeitung               | kurz bis mittel        | lang                | lang                |
| Strategieentwicklung       | sehr leistungsfähig    | sehr leistungsfähig | sehr leistungsfähig |
| Walk-Forward               | umfangreich            | umfangreich         | umfangreich         |
| Monte-Carlo                | integriert             | integriert          | integriert          |
| Noise-Tests                | integriert             | integriert          | integriert          |
| Codeexport                 | umfangreich            | gut                 | sehr gut            |
| Zielgruppe                 | Privat & Professionell | eher professionell  | eher professionell  |
| Preis-Leistungs-Verhältnis | ausgezeichnet          | gut                 | eher niedrig        |

*Die Bewertung basiert auf unseren eigenen praktischen Erfahrungen im Einsatz der jeweiligen Programme. Sie erhebt keinen Anspruch auf Allgemeingültigkeit.*

---

# Warum wir uns für Build Alpha entschieden haben

Alle drei Programme gehören zweifellos zu den leistungsfähigsten Genetic-Programming-Systemen, die derzeit am Markt verfügbar sind.

Nach vielen Monaten praktischer Nutzung haben wir uns dennoch entschieden, Build Alpha als offizieller Vertriebspartner im deutschsprachigen Raum anzubieten.

Dabei spielte nicht ein einzelnes Leistungsmerkmal die entscheidende Rolle.

Vielmehr überzeugte uns die Gesamtkombination aus

- einfacher Bedienung,
- sehr hoher Entwicklungsgeschwindigkeit,
- umfangreichen Robustheitstests,
- leistungsfähigem Codeexport
- und einem außergewöhnlich attraktiven Preis-Leistungs-Verhältnis.

Diese Einschätzung beruht nicht auf Herstellerunterlagen oder Marketingaussagen, sondern auf unserer eigenen praktischen Arbeit mit allen drei Plattformen.

Im nächsten Kapitel stellen wir Build Alpha deshalb ausführlich vor und zeigen anhand konkreter Beispiele, wie sich innerhalb kurzer Zeit professionelle Handelsstrategien entwickeln und bewerten lassen.

## Drei unterschiedliche Philosophien

- **GSB:** maximale Flexibilität
- **TSL Pro:** sehr großer Funktionsumfang
- **Build Alpha:** effiziente Strategieentwicklung mit intuitivem Workflow

# Kapitel 4

## Build Alpha in der Praxis

### Von der Idee zur fertigen Handelsstrategie

Die eigentliche Stärke von Build Alpha liegt nicht in einem einzelnen Indikator oder einer besonderen Optimierungsmethode.

Vielmehr überzeugt die Software durch einen durchgängigen Research-Workflow, der den gesamten Entwicklungsprozess automatisiert – von der Auswahl der Marktdaten bis zum fertigen Programmcode.

Während klassische Entwicklungsumgebungen häufig umfangreiche Programmierarbeiten oder manuelle Optimierungsschritte erfordern, verfolgt Build Alpha einen anderen Ansatz.

Der Anwender definiert den Rahmen der Untersuchung.

Die eigentliche Suche nach geeigneten Handelsstrategien übernimmt anschließend die Software.

Dadurch lassen sich innerhalb kurzer Zeit Hunderte oder sogar Tausende möglicher Strategien entwickeln und objektiv miteinander vergleichen.

---

## Schritt 1

### Auswahl des Marktes

Am Beginn jeder Untersuchung steht die Auswahl des Marktes.

Build Alpha unterstützt unter anderem

- Aktien
- Futures
- ETFs
- Devisen
- Kryptowährungen

Ebenso können verschiedene Zeiteinheiten untersucht werden – vom Intraday-Handel bis hin zu längerfristigen Swing- oder Positionstrading-Systemen.

Gerade professionelle Anwender schätzen diese Flexibilität, da sich identische Research-Prozesse auf unterschiedliche Märkte übertragen lassen.

---

## Definition des Suchraums

Anschließend wird festgelegt, in welchem Regeluniversum die Software nach Handelsstrategien suchen soll.

Hierzu gehören beispielsweise

- Einstiegssignale
- Ausstiegssignale
- Stop-Loss-Varianten
- Gewinnziele
- Zeitfilter
- Volatilitätsfilter
- Trendfilter
- saisonale Effekte
- Marktbreite
- Volumen
- ökonomische Einflussgrößen

Je nach Fragestellung können diese Komponenten beliebig kombiniert werden.

Dadurch entsteht ein sehr großer Suchraum möglicher Handelsstrategien.

## Schritt 3

### Automatische Strategiegenerierung

Nun beginnt der eigentliche Genetic-Programming-Prozess.

Build Alpha erzeugt automatisch eine große Anzahl vollständiger Handelssysteme.

Jede Strategie wird unmittelbar getestet und anhand zuvor definierter Fitness-Kriterien bewertet.

Zu den wichtigsten Kennzahlen gehören beispielsweise

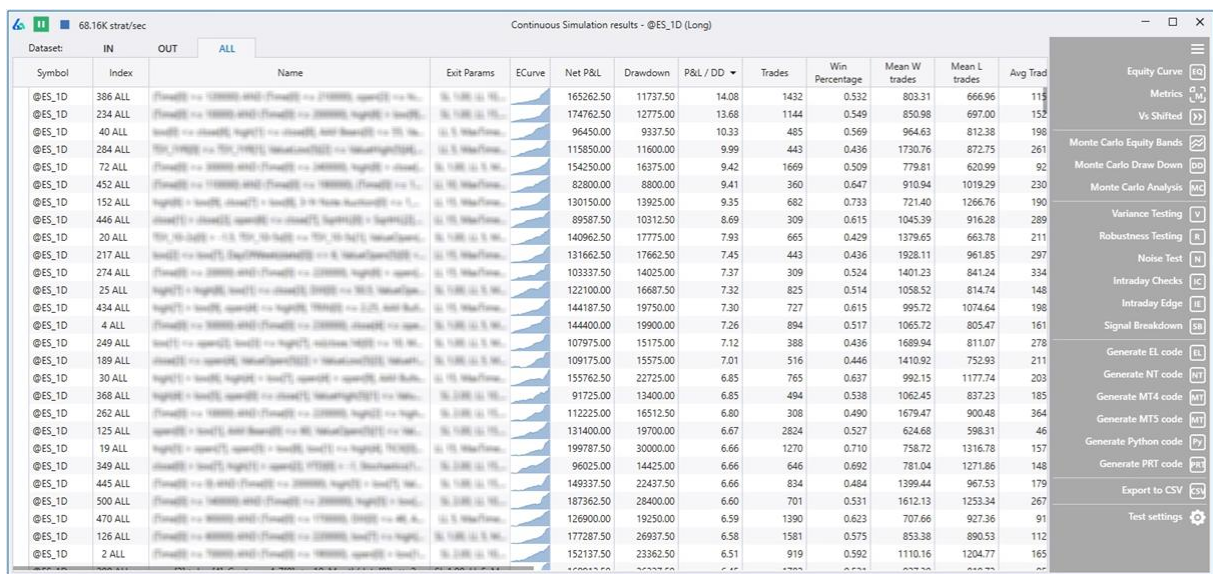
- Nettogewinn
- Profit Factor
- Sharpe Ratio
- Drawdown

- Gewinnwahrscheinlichkeit
- durchschnittlicher Trade
- Risiko-Rendite-Verhältnis
- Anzahl der Trades

Die besten Strategien werden anschließend weiterentwickelt.

Weniger erfolgreiche Varianten scheiden automatisch aus.

Dieser Prozess ähnelt biologischen Evolutionsmechanismen und führt häufig zu überraschenden Regelkombinationen, auf die ein Entwickler manuell kaum gekommen wäre.



| Symbol | Index   | Name                                                    | Exit Params        | ECurve | Net P&L   | Drawdown | P&L / DD | Trades | Win Percentage | Mean W trades | Mean L trades | Avg Trade |
|--------|---------|---------------------------------------------------------|--------------------|--------|-----------|----------|----------|--------|----------------|---------------|---------------|-----------|
| @ES_1D | 306 ALL | [Close] >= 10000 AND [Close] <= 10000, open() >= 0, ... | 0, 100, 0, 10, ... |        | 165262.50 | 11737.50 | 14.08    | 1432   | 0.532          | 803.31        | 666.96        | 118       |
| @ES_1D | 234 ALL | [Close] <= 10000 AND [Close] >= 10000, open() >= 0, ... | 0, 100, 0, 10, ... |        | 174762.50 | 12775.00 | 13.68    | 1144   | 0.549          | 850.98        | 697.00        | 152       |
| @ES_1D | 40 ALL  | [Close] >= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 96450.00  | 9337.50  | 10.33    | 485    | 0.569          | 964.63        | 812.38        | 198       |
| @ES_1D | 284 ALL | [Close] <= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 115850.00 | 11600.00 | 9.99     | 443    | 0.436          | 1730.76       | 872.75        | 261       |
| @ES_1D | 72 ALL  | [Close] >= 10000 AND [Close] <= 10000, open() >= 0, ... | 0, 100, 0, 10, ... |        | 154250.00 | 16375.00 | 9.42     | 1669   | 0.509          | 779.81        | 620.99        | 92        |
| @ES_1D | 452 ALL | [Close] <= 10000 AND [Close] >= 10000, open() >= 0, ... | 0, 100, 0, 10, ... |        | 82800.00  | 8800.00  | 9.41     | 360    | 0.647          | 910.94        | 1019.29       | 230       |
| @ES_1D | 152 ALL | [Close] >= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 130150.00 | 13925.00 | 9.35     | 682    | 0.733          | 721.40        | 1266.76       | 190       |
| @ES_1D | 446 ALL | [Close] <= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 89587.50  | 10312.50 | 8.69     | 309    | 0.615          | 1045.39       | 916.28        | 289       |
| @ES_1D | 20 ALL  | [Close] >= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 140962.50 | 17775.00 | 7.93     | 665    | 0.429          | 1379.65       | 663.78        | 211       |
| @ES_1D | 217 ALL | [Close] <= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 131662.50 | 17662.50 | 7.45     | 443    | 0.436          | 1928.11       | 961.85        | 297       |
| @ES_1D | 274 ALL | [Close] >= 10000 AND [Close] <= 10000, open() >= 0, ... | 0, 100, 0, 10, ... |        | 103337.50 | 14025.00 | 7.37     | 309    | 0.534          | 1401.23       | 841.24        | 334       |
| @ES_1D | 25 ALL  | [Close] <= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 122100.00 | 16687.50 | 7.32     | 825    | 0.514          | 1058.52       | 814.74        | 148       |
| @ES_1D | 434 ALL | [Close] >= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 144187.50 | 19750.00 | 7.30     | 727    | 0.615          | 995.72        | 1074.64       | 198       |
| @ES_1D | 4 ALL   | [Close] <= 10000 AND [Close] >= 10000, open() >= 0, ... | 0, 100, 0, 10, ... |        | 144400.00 | 19900.00 | 7.26     | 894    | 0.517          | 1065.72       | 805.47        | 161       |
| @ES_1D | 249 ALL | [Close] >= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 107975.00 | 15175.00 | 7.12     | 388    | 0.436          | 1689.94       | 811.07        | 278       |
| @ES_1D | 189 ALL | [Close] <= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 109175.00 | 15575.00 | 7.01     | 516    | 0.446          | 1410.92       | 752.93        | 211       |
| @ES_1D | 30 ALL  | [Close] >= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 155762.50 | 22725.00 | 6.85     | 765    | 0.637          | 992.15        | 1177.74       | 203       |
| @ES_1D | 368 ALL | [Close] <= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 91725.00  | 13400.00 | 6.85     | 494    | 0.538          | 1062.45       | 837.23        | 185       |
| @ES_1D | 262 ALL | [Close] >= 10000 AND [Close] <= 10000, open() >= 0, ... | 0, 100, 0, 10, ... |        | 112225.00 | 16512.50 | 6.80     | 308    | 0.490          | 1679.47       | 900.48        | 364       |
| @ES_1D | 125 ALL | [Close] <= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 131400.00 | 19700.00 | 6.67     | 2824   | 0.527          | 624.68        | 598.31        | 46        |
| @ES_1D | 19 ALL  | [Close] >= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 199787.50 | 30000.00 | 6.66     | 1270   | 0.710          | 758.72        | 1316.78       | 157       |
| @ES_1D | 349 ALL | [Close] <= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 96025.00  | 14425.00 | 6.66     | 646    | 0.692          | 781.04        | 1271.86       | 148       |
| @ES_1D | 445 ALL | [Close] >= 10000 AND [Close] <= 10000, open() >= 0, ... | 0, 100, 0, 10, ... |        | 149337.50 | 22437.50 | 6.66     | 834    | 0.484          | 1399.44       | 967.53        | 179       |
| @ES_1D | 500 ALL | [Close] <= 10000, open() >= 0, ...                      | 0, 100, 0, 10, ... |        | 187962.50 | 28400.00 | 6.60     | 701    | 0.531          | 1612.13       | 1253.34       | 267       |
| @ES_1D | 470 ALL | [Close] >= 10000 AND [Close] <= 10000, open() >= 0, ... | 0, 100, 0, 10, ... |        | 126900.00 | 19250.00 | 6.59     | 1390   | 0.623          | 707.66        | 927.36        | 91        |
| @ES_1D | 126 ALL | [Close] <= 10000 AND [Close] >= 10000, open() >= 0, ... | 0, 100, 0, 10, ... |        | 177287.50 | 26937.50 | 6.58     | 1581   | 0.575          | 853.38        | 890.53        | 112       |
| @ES_1D | 2 ALL   | [Close] >= 10000 AND [Close] <= 10000, open() >= 0, ... | 0, 100, 0, 10, ... |        | 152137.50 | 23362.50 | 6.51     | 919    | 0.592          | 1110.16       | 1204.77       | 165       |

## Schritt 4

### Objektive Robustheitsprüfung

Ein überzeugender Backtest allein reicht heute nicht mehr aus. Professionelle Strategien müssen auch unter veränderten Marktbedingungen bestehen.

Build Alpha stellt hierfür umfangreiche Validierungsverfahren zur Verfügung.

Dazu gehören unter anderem

- Walk-Forward-Analysen
- Monte-Carlo-Simulationen
- Noise-Tests
- Out-of-Sample-Analysen
- Parameterstabilität
- Forward-Variance-Tests



Gerade dieser Schritt entscheidet häufig darüber, ob eine Strategie lediglich gut aussieht oder tatsächlich robust genug für einen praktischen Einsatz ist.

Nach unserer Erfahrung gehört dieser Bereich zu den größten Stärken von Build Alpha.

INFOBOX 03

Typische Fehler bei der Strategieentwicklung

X Zu wenige Trades

X Zu viele Parameter

X Überoptimierung

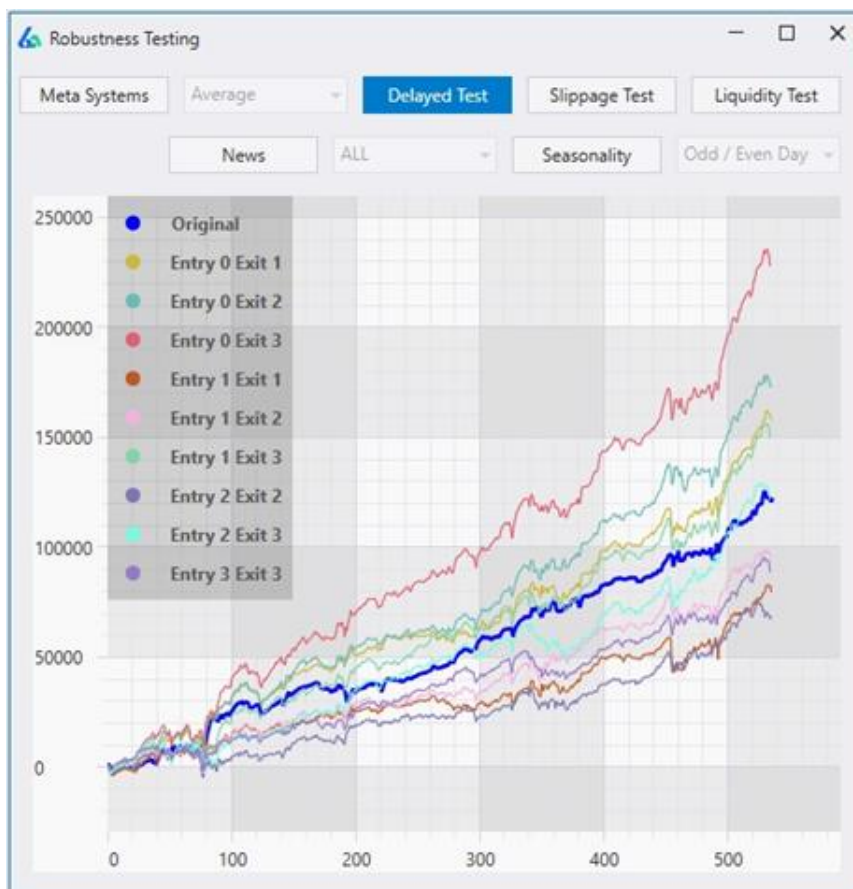
X Keine Out-of-Sample-Tests

X Keine Monte-Carlo-Analyse

X Zu starke Korrelation im Portfolio

Genetic Programming löst diese Probleme nicht automatisch.

Es hilft jedoch, robuste Strategien deutlich schneller zu identifizieren.



## Schritt 5

### Analyse der Ergebnisse

Nach Abschluss der Berechnungen stellt Build Alpha sämtliche Strategien in übersichtlichen Tabellen dar.

Der Anwender kann die Ergebnisse nach nahezu beliebigen Kriterien sortieren und filtern.

Dadurch wird es möglich,

- Strategien miteinander zu vergleichen,
- Korrelationen zu erkennen,
- geeignete Portfolio-Kandidaten auszuwählen,
- und robuste Systeme gezielt weiterzuentwickeln.

Die eigentliche Forschungsarbeit beginnt häufig erst an diesem Punkt. Nicht jede profitable Strategie eignet sich automatisch für den Live-Handel.

Erfahrung und kritische Analyse bleiben deshalb weiterhin unverzichtbar.

```
+-----+
| INFOBOX 04 |
| |
| Warum ein guter Backtest nicht genügt |
| |
| Ein Backtest beantwortet nur eine Frage: |
| |
| "Wie hätte die Strategie in der Vergangenheit |
| funktioniert?" |
| |
| Entscheidend sind jedoch weitere Fragen: |
| |
| ✓ Bleibt sie auch morgen stabil? |
| ✓ Reagiert sie empfindlich auf Parameter? |
| ✓ Besteht sie Walk-Forward-Tests? |
| ✓ Übersteht sie Monte-Carlo-Simulationen? |
| |
| Erst danach beginnt die eigentliche Bewertung. |
+-----+
```

## Schritt 6

### Codeexport

Sobald eine Strategie überzeugt, erzeugt Build Alpha automatisch lauffähigen Programmcode.

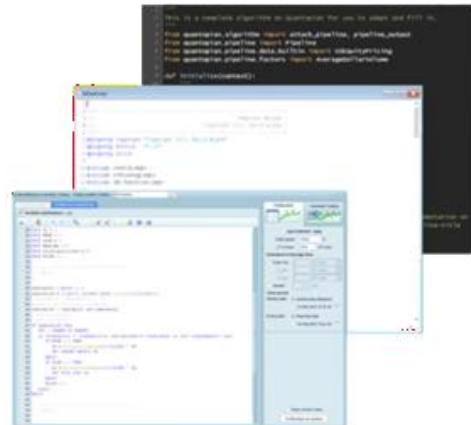
Unterstützt werden unter anderem

- TradeStation EasyLanguage
- MultiCharts PowerLanguage
- NinjaTrader
- MetaTrader
- Python

Dadurch entfällt in vielen Fällen das zeitaufwendige manuelle Programmieren.

Die Strategien können unmittelbar überprüft, weiterentwickelt oder in bestehende Handelsumgebungen integriert werden.

Gerade für professionelle Research-Abteilungen bedeutet dies eine erhebliche Zeitersparnis.



## Unsere Erfahrungen

Wir haben Build Alpha nicht deshalb ausgewählt, weil einzelne Funktionen exklusiv wären.

Viele professionelle Programme verfügen heute über ähnliche Möglichkeiten. Entscheidend war für uns vielmehr die Gesamtheit des Entwicklungsprozesses.

Build Alpha führt den Anwender in einer klar strukturierten Reihenfolge durch sämtliche Schritte der Strategieentwicklung. Dadurch bleibt der Fokus auf dem eigentlichen Ziel erhalten:

**Robuste Handelsstrategien zu entwickeln – nicht Software zu konfigurieren.**

Gerade dieser effiziente Workflow unterscheidet Build Alpha aus unserer Sicht von vielen anderen professionellen Entwicklungsumgebungen.



---

## Zusammenfassung

Build Alpha automatisiert nicht nur einzelne Arbeitsschritte.

Die Software bildet den kompletten Entwicklungsprozess professioneller Handelssysteme ab – von der ersten Idee bis zum fertigen Programmcode.

Dadurch können sich quantitative Trader und Research-Abteilungen auf den entscheidenden Teil ihrer Arbeit konzentrieren:

**Die objektive Bewertung und Auswahl robuster Handelsstrategien.**

## Kapitel 5

# Unsere praktischen Erfahrungen mit Build Alpha

Software lässt sich anhand von Funktionslisten vergleichen.



Ob sie sich im täglichen Einsatz bewährt, zeigt sich jedoch erst nach vielen Monaten praktischer Arbeit.

Aus diesem Grund haben wir verschiedene Genetic-Programming-Plattformen nicht nur oberflächlich getestet, sondern intensiv im Rahmen unserer täglichen Forschungsarbeit eingesetzt.

Unser Ziel bestand dabei nicht darin, möglichst viele Strategien zu erzeugen. Uns interessierte vielmehr die Frage:

**Welche Software unterstützt einen professionellen Entwicklungsprozess am effektivsten?**

Gerade bei institutionellen Anwendungen stehen nicht möglichst spektakuläre Backtests im Vordergrund. Entscheidend sind vielmehr reproduzierbare Ergebnisse, robuste Strategien und ein effizienter Research-Workflow.

---

## Forschung statt Optimierung

Ein häufiger Fehler bei der Entwicklung automatischer Handelssysteme besteht darin, möglichst lange an einer einzelnen Strategie zu arbeiten.

- Neue Filter werden ergänzt.
- Parameter werden verändert.
- Weitere Optimierungen werden durchgeführt.

Irgendwann entsteht der Eindruck, eine besonders gute Strategie entwickelt zu haben. In vielen Fällen wurde jedoch lediglich die Vergangenheit immer genauer beschrieben.

Build Alpha verfolgt einen anderen Ansatz.

Anstatt sehr viel Zeit in einzelne Strategien zu investieren, ermöglicht die Software die Entwicklung einer großen Anzahl unterschiedlicher Handelsmodelle.

Dadurch verschiebt sich der Schwerpunkt der Arbeit. Nicht die Optimierung einzelner Ideen steht im Mittelpunkt. Entscheidend wird vielmehr die objektive Auswahl wirklich robuster Strategien.

Nach unserer Erfahrung führt genau dieser Perspektivwechsel häufig zu deutlich besseren Ergebnissen.



## Zeit ist einer der wichtigsten Faktoren

Professionelles Research bedeutet nicht nur, gute Strategien zu entwickeln. Ebenso wichtig ist die Geschwindigkeit, mit der neue Ideen überprüft werden können.

Viele Entwicklungsumgebungen bieten einen enormen Funktionsumfang, erfordern jedoch zahlreiche manuelle Konfigurationsschritte. Dadurch vergeht häufig viel Zeit, bevor überhaupt erste Ergebnisse vorliegen.

Build Alpha verfolgt einen wesentlich direkteren Workflow.

Nach Auswahl des Marktes und der gewünschten Rahmenbedingungen beginnt die eigentliche Strategiegenerierung unmittelbar. Gerade dieser effiziente Ablauf hat uns im täglichen Einsatz besonders überzeugt.

Nicht weil einzelne Funktionen exklusiv wären. Sondern weil sich der gesamte Entwicklungsprozess deutlich flüssiger anfühlt.

---

## Die eigentliche Arbeit beginnt nach dem Backtest

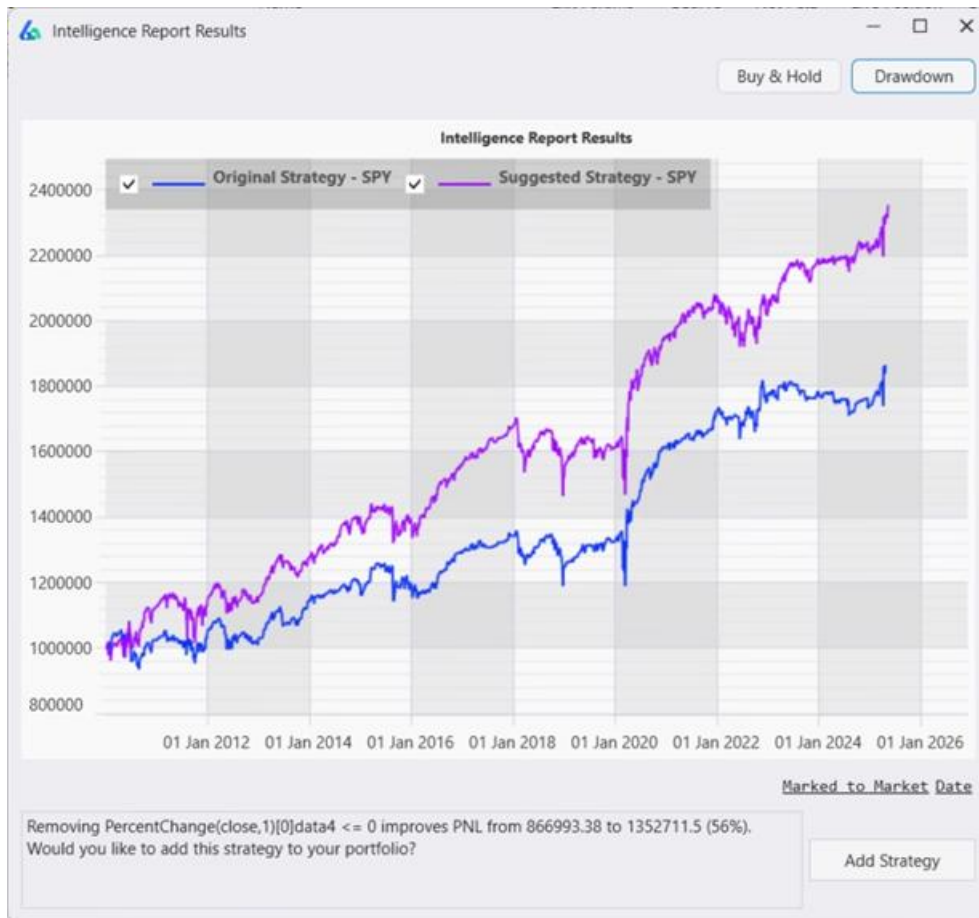
Viele Einsteiger betrachten einen erfolgreichen Backtest bereits als fertige Handelsstrategie. Unsere Erfahrung ist eine andere.

Ein Backtest ist lediglich der Ausgangspunkt weiterer Untersuchungen.

Erst anschließend beginnt die eigentliche Bewertung. Dabei interessieren unter anderem folgende Fragen:

- Bleibt die Strategie auch außerhalb des Trainingszeitraums stabil?
- Wie empfindlich reagiert sie auf kleine Parameteränderungen?
- Wie entwickelt sich die Performance unter Monte-Carlo-Simulationen?
- Welche Auswirkungen haben zufällige Störungen der Kursdaten?
- Ergänzt die Strategie ein bestehendes Portfolio oder erhöht sie lediglich das Risiko?

Gerade in dieser Phase liefert Build Alpha umfangreiche Hilfsmittel, um robuste Strategien von zufälligen Ergebnissen zu unterscheiden.



---

## Portfolio statt Einzelstrategie

In der Vergangenheit konzentrierte sich die Entwicklung automatischer Handelssysteme häufig auf einzelne Märkte. Heute betrachten professionelle Anwender zunehmend das Gesamtportfolio.

Eine durchschnittliche Strategie kann innerhalb eines gut diversifizierten Portfolios wertvoller sein als eine hervorragende Einzelstrategie, deren Ergebnisse stark mit anderen Systemen korrelieren. Auch deshalb spielt die Portfolioanalyse im modernen quantitativen Research eine immer größere Rolle.

Nach unserer Erfahrung unterstützt Build Alpha diesen Denkansatz sehr konsequent. Nicht die einzelne Strategie steht im Mittelpunkt. Entscheidend ist der Beitrag zum gesamten Portfolio.

---



## Unsere Bewertung

Nach intensiver praktischer Nutzung verschiedener Genetic-Programming-Systeme kommen wir zu folgendem Ergebnis.

Alle untersuchten Programme ermöglichen die Entwicklung leistungsfähiger Handelssysteme. Die Unterschiede liegen weniger in der grundsätzlichen Qualität der Ergebnisse als vielmehr im Arbeitsablauf.

Gerade hier sehen wir die größte Stärke von Build Alpha.

Die Software verbindet einen sehr großen Funktionsumfang mit einem klar strukturierten Workflow und ermöglicht dadurch einen besonders effizienten Entwicklungsprozess.

Diese Einschätzung beruht ausdrücklich auf unseren eigenen praktischen Erfahrungen. Sie stellt keine allgemeingültige Bewertung dar. Je nach Anforderungen kann auch eine andere Plattform die geeignete Wahl sein.

Für unsere tägliche Forschungsarbeit hat sich Build Alpha jedoch als das Werkzeug erwiesen, das unsere Anforderungen insgesamt am besten erfüllt.

---

## Worauf wir bei einer Research-Plattform besonders achten

- ✓ Strukturiertes Workflow
- ✓ Hohe Entwicklungsgeschwindigkeit
- ✓ Aussagekräftige Robustheitstests
- ✓ Gute Vergleichbarkeit vieler Strategien
- ✓ Portfolioorientiertes Arbeiten
- ✓ Zuverlässiger Codeexport
- ✓ Angemessenes Preis-Leistungs-Verhältnis

## Kapitel 6

### Von der Strategie zum Echtgeldportfolio

Eine Handelsstrategie allein stellt noch keinen nachhaltigen Anlageprozess dar.

Erst das Zusammenspiel mehrerer möglichst unabhängig arbeitender Strategien bildet die Grundlage eines robusten systematischen Portfolios.

Aus diesem Grund betrachten wir Genetic Programming nicht als Werkzeug zur Entwicklung einzelner Handelssysteme, sondern als Research-Plattform zur Entwicklung kompletter Portfolios.

Gerade hierin sehen wir einen der größten Vorteile moderner Genetic-Programming-Systeme.

---

## Diversifikation beginnt bereits im Research

Diversifikation bedeutet nicht nur, verschiedene Märkte gleichzeitig zu handeln. Ebenso wichtig ist die Diversifikation der Handelslogiken.

Zwei Strategien auf unterschiedlichen Märkten können dennoch nahezu identisch reagieren. Umgekehrt können mehrere Strategien auf demselben Markt vollkommen unterschiedliche Eigenschaften besitzen.

Deshalb untersuchen wir während des Entwicklungsprozesses nicht nur die Qualität einzelner Strategien, sondern auch deren Zusammenspiel innerhalb eines Gesamtportfolios.

Dabei interessieren unter anderem folgende Fragen:

- Ergänzen sich die Strategien gegenseitig?
- Wie stark korrelieren ihre Ergebnisse?
- Welche Strategie trägt welchen Anteil zum Gesamtrisiko bei?
- Wie verändert sich der maximale Drawdown?
- Welche Auswirkungen ergeben sich auf die Stabilität der Eigenkapitalkurve?

Erst wenn diese Fragen zufriedenstellend beantwortet sind, betrachten wir eine Strategie als potenziellen Bestandteil eines Echtgeldportfolios.

## Ein Praxisbeispiel

Die nachfolgenden Ergebnisse stammen aus einem Echtgeldportfolio, dessen Handelsstrategien mit modernen Genetic-Programming-Werkzeugen entwickelt wurden.

Zum Einsatz kamen unter anderem Strategien für folgende CME Micro Futures:

- MNQ – Micro E-mini Nasdaq-100
- MES – Micro E-mini S&P 500
- MYM – Micro E-mini Dow Jones
- M2K – Micro Russell 2000
- MGC – Micro Gold
- MCL – Micro WTI Crude Oil
- M6E – Micro EUR/USD
- SIL – Micro Silver



Das Portfolio umfasst rund **45 Handelsstrategien**, die jeweils mit zwei Micro-Kontrakten gehandelt wurden.

Im dargestellten Zeitraum wurden **weit über 1.500 reale Transaktionen** ausgeführt.

Diese hohe Anzahl realer Trades ermöglicht eine wesentlich aussagekräftigere statistische Beurteilung als Systeme mit wenigen Einzeltransaktionen.

**Wichtig:** Die dargestellten Strategien wurden mit modernen Genetic-Programming-Werkzeugen entwickelt, darunter auch Build Alpha.



Ein wesentlich größeres Portfolio wurde in 2025 ausschließlich mit Build Alpha entwickelt und kam 2026 in den Realhandel. Es handelt die liquidesten Futures verschiedener Asset Klassen (Aktien-Index, Währungen, Bonds, Commodities usw.). **Aktuelle Auswertungen auf Anfrage !**

## Warum Micro Futures?

Die Einführung der CME Micro Futures hat den Aufbau diversifizierter Portfolios erheblich erleichtert. Bereits mit vergleichsweise überschaubarem Kapital lassen sich heute mehrere Märkte parallel handeln.

Dadurch können Risiken auf unterschiedliche Anlageklassen verteilt werden. Gleichzeitig wird es möglich, neue Strategien zunächst mit geringer Positionsgröße unter realen Marktbedingungen zu validieren.

Gerade für quantitative Entwickler eröffnet dies völlig neue Möglichkeiten.



# Von der Forschung in den Live-Handel

Zwischen einer erfolgreichen Strategie im Backtest und ihrem Einsatz im Echtgeldhandel liegen mehrere zusätzliche Schritte.

Nach unserer Vorgehensweise durchläuft jede Strategie unter anderem folgende Phasen:

1. Entwicklung mittels Genetic Programming
2. Auswahl anhand objektiver Qualitätskriterien
3. Umfangreiche Robustheitstests
4. Integration in das Portfolio
5. Paper Trading
6. Laufende Überwachung
7. Echtgeldhandel

Erst wenn sämtliche Prüfschritte erfolgreich abgeschlossen wurden, erfolgt die Aufnahme in ein Echtgeldportfolio. Dieser Prozess ist zeitaufwendig. Er reduziert jedoch das Risiko, statistisch zufällige Ergebnisse mit dauerhaft belastbaren Handelsvorteilen zu verwechseln.

---

## Research ist ein kontinuierlicher Prozess

- Märkte verändern sich.
- Neue Marktteilnehmer treten auf.
- Liquidität verschiebt sich.
- Regulatorische Rahmenbedingungen ändern sich.
- Deshalb verstehen wir quantitative Forschung nicht als einmalige Entwicklung einer Handelsstrategie.
- Research ist ein kontinuierlicher Prozess.
- Neue Ideen entstehen fortlaufend.
- Bestehende Strategien werden regelmäßig überprüft.
- Nicht mehr robuste Systeme werden ersetzt.
- Neue Ansätze ergänzen das Portfolio.

Gerade hierbei unterstützt Build Alpha den gesamten Entwicklungsprozess auf überzeugende Weise.

---

## Unsere Erfahrung

Die größte Stärke moderner Genetic-Programming-Plattformen besteht aus unserer Sicht nicht darin, möglichst viele Strategien zu erzeugen. Entscheidend ist vielmehr, den gesamten Prozess der Ideenfindung erheblich zu beschleunigen.

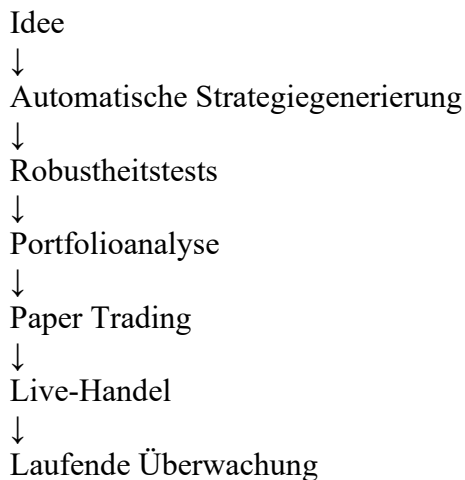
Dadurch können quantitative Entwickler ihre Zeit auf die wirklich wichtigen Aufgaben konzentrieren:

- Bewertung der Ergebnisse
- Risikomanagement
- Portfoliozusammenstellung
- statistische Validierung
- kontinuierliche Weiterentwicklung bestehender Strategien

Genau hierin sehen wir den eigentlichen Nutzen von Build Alpha.

---

## Unser Entwicklungsprozess



---

## Zusammenfassung

Professionelle quantitative Forschung endet nicht mit einem erfolgreichen Backtest.

Erst die Kombination aus objektiver Strategiegenerierung, konsequenter Robustheitsprüfung und sinnvoller Portfoliozusammenstellung schafft die Grundlage für einen langfristig stabilen Handelsprozess.

Build Alpha unterstützt diesen gesamten Workflow und hat sich deshalb als fester Bestandteil unseres täglichen Research-Prozesses etabliert.



## Kapitel 7

### Warum Build Alpha über WST?

Build Alpha ist eine außergewöhnlich leistungsfähige Software. Den eigentlichen Mehrwert sehen wir jedoch nicht allein im Programm selbst.

Entscheidend ist aus unserer Sicht, wie effizient sich die Software im täglichen Research einsetzen lässt. Genau hier möchten wir unsere Kunden unterstützen.

Wir verstehen uns nicht als klassischer Softwarehändler. Wir verkaufen keine Software "von der Stange", sondern begleiten unsere Kunden bei der Einführung einer neuen Research-Methode !

### Mehr als drei Jahrzehnte Erfahrung

Die WST – Wall Street Technology GmbH beschäftigt sich seit mehr als 30 Jahren mit der Entwicklung quantitativer Handelsstrategien.

Bereits in den 1990er Jahren untersuchten wir Verfahren der Künstlichen Intelligenz und des Maschinellen Lernens zur Entwicklung systematischer Handelssysteme.

Im Laufe der Jahre kamen zahlreiche weitere Methoden hinzu, darunter:

- Genetic Programming
- Evolutionäre Optimierung
- Portfoliooptimierung
- Walk-Forward-Analysen
- Monte-Carlo-Verfahren
- Statistische Validierung

Diese Erfahrungen fließen heute unmittelbar in unsere tägliche Forschungsarbeit ein.

---



# Was unsere Kunden erhalten

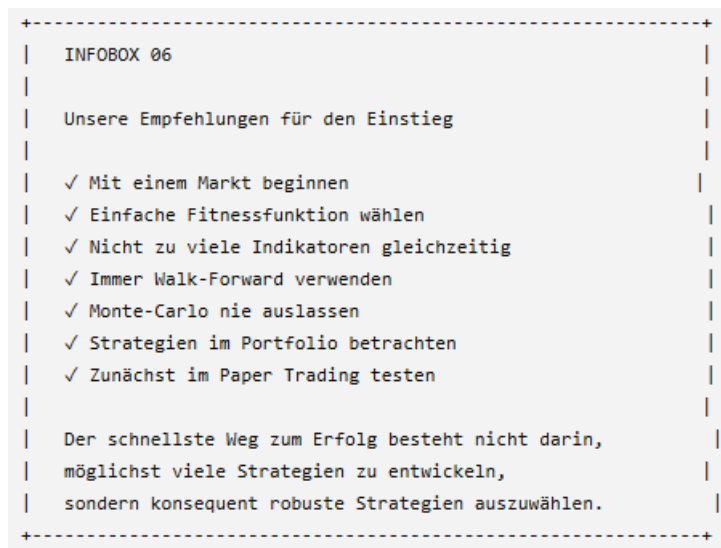
Beim Erwerb einer Build-Alpha-Lizenz über WST profitieren unsere Kunden von mehreren Leistungen, die über die eigentliche Software hinausgehen.

## Unterstützung bei

- Installation
- Einrichtung
- Lizenzaktivierung
- ersten Research-Projekten
- Codeexport
- Interpretation der Ergebnisse

Darüber hinaus stehen wir bei Fragen zur praktischen Anwendung als Ansprechpartner zur Verfügung.

Selbstverständlich bleibt auch der direkte Support des Herstellers weiterhin erhalten.



## Unser Anspruch

Unser Ziel besteht nicht darin, möglichst viele Strategien zu erzeugen.

Wir möchten unsere Kunden dabei unterstützen,

- robuste Research-Prozesse aufzubauen,
- statistisch belastbare Ergebnisse zu erhalten,
- typische Fehler bei der Strategieentwicklung zu vermeiden,
- und Genetic Programming effizient in den eigenen Entwicklungsprozess zu integrieren.

Gerade dieser Wissenstransfer ist aus unserer Sicht mindestens ebenso wichtig wie die Software selbst.

---

## Für wen eignet sich Build Alpha?

Build Alpha richtet sich an Anwender, die systematisch und datenbasiert arbeiten möchten. Dazu gehören insbesondere:

### **Quantitative Trader**

Entwicklung neuer Handelsstrategien und Portfolios.

### **Vermögensverwalter**

Suche nach zusätzlichen Alpha-Quellen und diversifizierten Handelsansätzen.

### **CTA und Hedgefonds**

Beschleunigung des internen Research-Prozesses.

### **Family Offices**

Aufbau regelbasierter Investmentprozesse.

### **Ambitionierte Privatanleger**

Professionelle Strategieentwicklung ohne umfangreiche Programmierkenntnisse.



## **Hochschulen und Forschungseinrichtungen**

Untersuchung moderner Verfahren des Genetic Programming und der quantitativen Finanzforschung.

---

## **Zusammenfassung**

Build Alpha bietet eine leistungsfähige Plattform zur automatisierten Entwicklung regelbasierter Handelsstrategien.

Durch den Erwerb über WST erhalten unsere Kunden zusätzlich einen deutschsprachigen Ansprechpartner mit langjähriger Erfahrung im quantitativen Research.

Unser Ziel besteht nicht darin, lediglich Software zu verkaufen.

Wir möchten unsere Kunden dabei unterstützen, Genetic Programming erfolgreich in ihren eigenen Entwicklungsprozess zu integrieren.

## **Fazit**

### **Genetic Programming verändert die Entwicklung automatischer Handelssysteme**

Die Entwicklung erfolgreicher Handelsstrategien war schon immer ein anspruchsvoller Forschungsprozess.

Steigende Datenmengen, leistungsfähigere Computer und moderne Verfahren der Künstlichen Intelligenz haben diesen Prozess in den vergangenen Jahren grundlegend verändert.

Anstatt einzelne Handelsideen manuell zu entwickeln und anschließend zu optimieren, ermöglichen moderne Genetic-Programming-Systeme heute die automatisierte Entwicklung und objektive Bewertung tausender unterschiedlicher Handelsstrategien.

Dadurch entstehen neue Möglichkeiten für quantitative Trader, Vermögensverwalter und institutionelle Research-Abteilungen.

Der eigentliche Vorteil liegt dabei nicht allein in der höheren Geschwindigkeit.

Vielmehr wird der gesamte Entwicklungsprozess systematischer, reproduzierbarer und besser nachvollziehbar.



Professionelle Entwickler können sich dadurch auf die eigentliche Forschungsarbeit konzentrieren:

- Entwicklung neuer Handelsideen
- Bewertung der Ergebnisse
- statistische Validierung
- Portfoliozusammenstellung
- kontinuierliche Verbesserung bestehender Strategien

---

## Build Alpha kennenlernen

Wenn Sie erfahren möchten,

- wie Build Alpha arbeitet,
- wie Genetic Programming Ihren Research-Prozess beschleunigen kann,
- oder welche Möglichkeiten sich für Ihre eigenen Handelsstrategien ergeben,

stehen wir Ihnen gerne für ein persönliches Gespräch oder eine Online-Demonstration zur Verfügung.

**[Bestellung von Build Alpha mit deutschsprachigem Support über WST GmbH NUR über diesen Link !!!](https://www.paypal.com/ncp/payment/LVSBAJQFJE64U)** <https://www.paypal.com/ncp/payment/LVSBAJQFJE64U>

---

## Kontakt

### WST – Wall Street Technology GmbH

Michael Kühn

 [www.wallst.de](http://www.wallst.de)

✉ [kuehn@wallst.de](mailto:kuehn@wallst.de)

☎ +49 30 3235522

---

## Schlusswort

**"Erfolgreiche quantitative Forschung beginnt nicht mit der Optimierung einer Strategie – sondern mit einem strukturierten Entwicklungsprozess."**