

# Quero

**The New Web Experience**

<http://www.quero.at/>

DI Viktor Krammer

# Quero

---

Qui quaerit, invenit.



Biblia Vulgata, Lc 11, 9

---

# Quero **Zur Person: Viktor Krammer**

---

- Seit **2004** Researcher im Forschungsverein EC3 E-Commerce Competence Center Web-Mining und Data-Warehousing
- **1999-2004** Informatikstudium an der TU Wien
- **2001-2004** IT-Freelancer für Frequentis, Siemens Österreich, Bull
- Seit **2004** Entwicklung Quero Toolbar Suchmaschinen-unabhängige Umsetzung des Forschungsprojekts (Dissertation an der TU Wien) als eigenes Produkt

# Quero Agenda

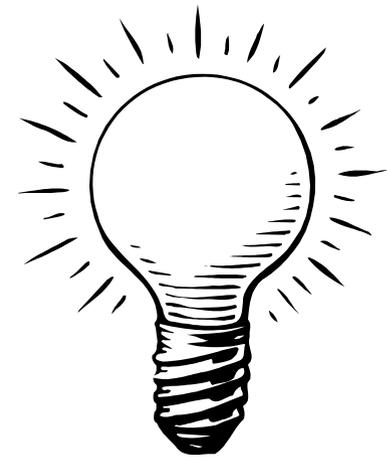
---

- **Innovation Quero Toolbar**  
Usability, Security, Privacy
- **Technologie**  
Windows Internet Explorer, Architektur
- **Softwarequalität**  
Entwicklungsmodell, Qualitätsmaßnahmen,  
Herausforderungen

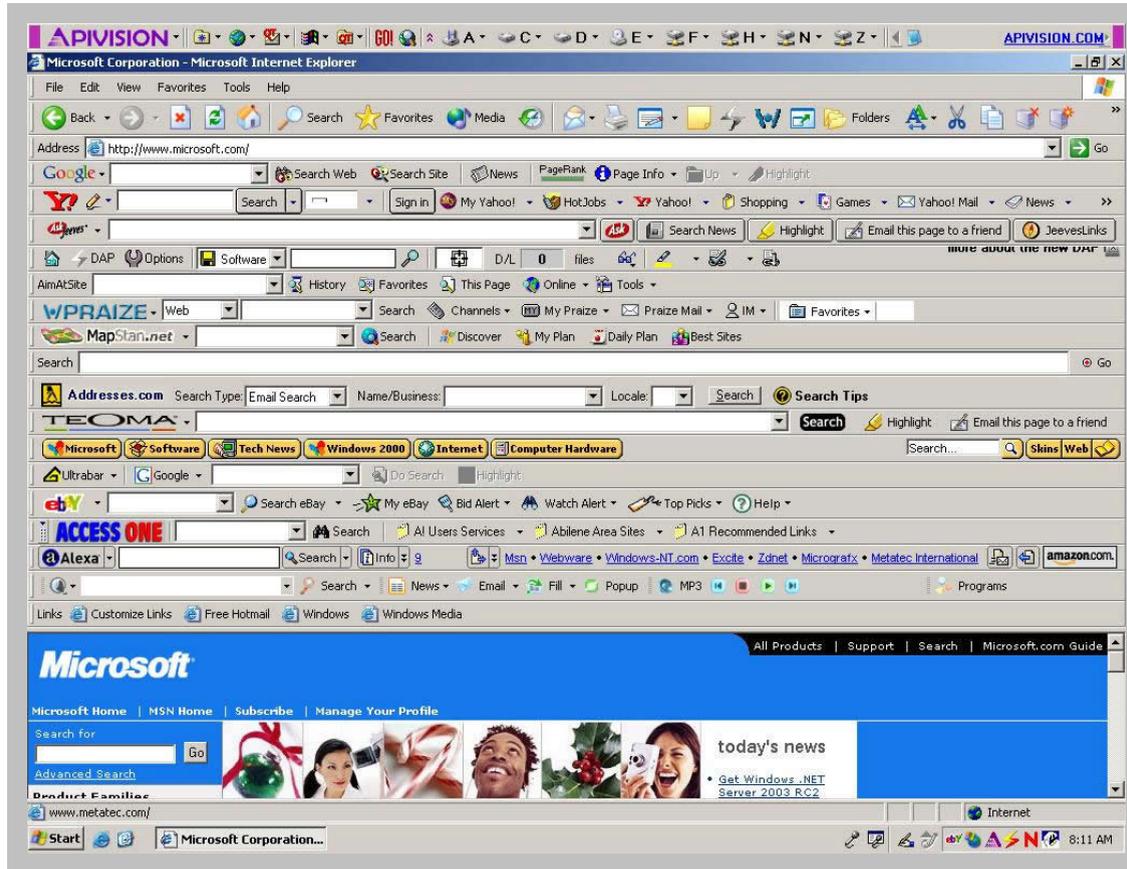
# Quero Die Idee

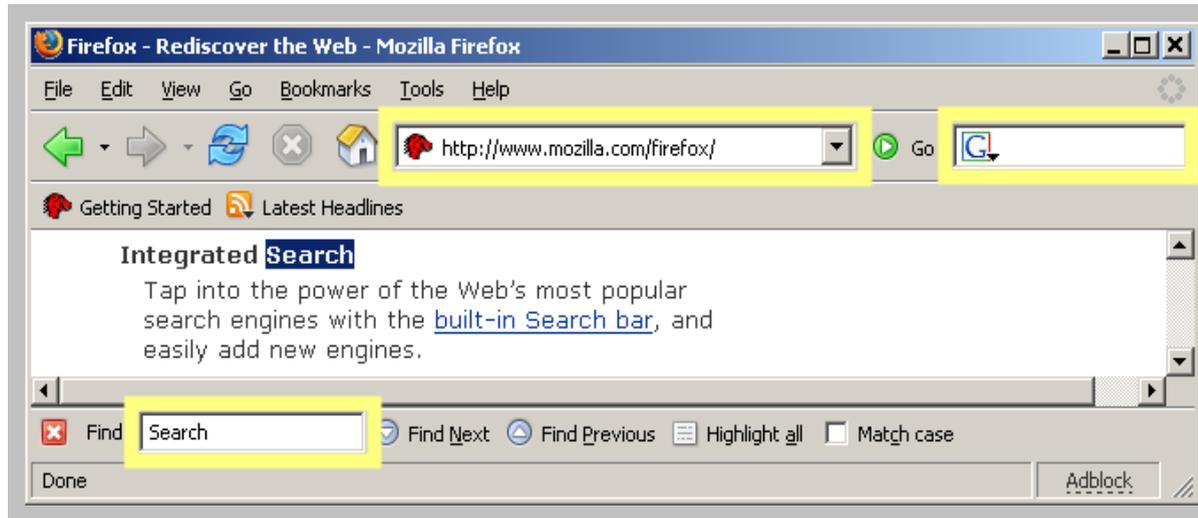
---

- Suchen direkt aus dem Browser
- Innovatives Userinterface
- Neue Funktionen für den IE
- Mehr Sicherheit beim Web-Surfen
- Opt-Out für Online-Werbung
- Privacy Inside



# Quero Toolbar-Chaos im IE6





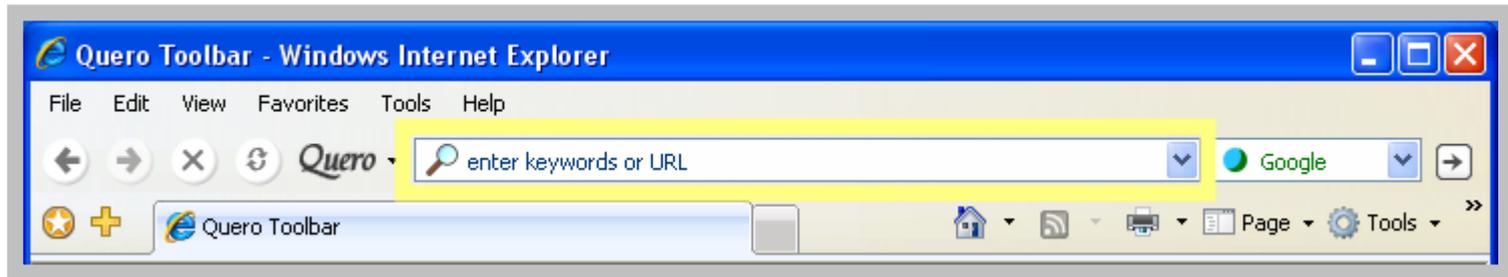
- Mehrere Eingabefelder
- Suchfeld muss Platz mit der Adressleiste teilen
- Eigene Tastatur-Shortcuts für jedes Feld

# Quero Lösung der Quero Toolbar

---

1 Eingabefeld für 3 Anwendungsfälle:

- Navigation (Adressleiste)
- Suche im Web (Suchleiste)
- Suche auf der aktuellen Seite (Dialog Suchen im IE)



# Quero **Wie funktioniert Quero?**

---

Syntaktische Unterscheidung der Suchtypen

- Schlüsselwort **ORF**
- Webadresse **ORF.at**
- Wortsuche **/orf**

Komplexe Suchanfragen

- ÖBB-Zugverbindung **Wien – Linz**



# Quero Internationale Domain Namen

---

- Hierzulande auch als „Umlautdomänen“ bekannt

Beispiel: [www.oberoesterreich.at](http://www.oberoesterreich.at)

- RFC-konforme Implementierung des IDNA-Standards (Internationalized Domain Names in Applications)

- Schutz vor Address-Spoofing-Attacken

Beispiel: Homograph-Attacke

<http://www.paypal.com/>

a ... U+0061 Latin Small Letter a

a ... U+0430 Cyrillic Small Letter a



Phishing

# Quero Address-Spoofing-Schutz

---

- Visualisierungstechniken
    - Hervorheben von Zeichen, die einem anderen Zeichensatz angehören
    - Hervorheben der „Kern“-Domäne
    - IDN-Symbol und Ziffern-Symbol z.B. bei [www.microsoft.com](http://www.microsoft.com)
    - Schloss-Symbol für sichere Verbindungen
  - UI-Verbesserungen
  - Sicherheitswarnung bei besonders verdächtigen Adressen
-

# Quero Opt-Out für Online-Werbung

---

- Pop-Up-Blocker
- Ausblenden von Flash-Werbung bei Bedarf
- Content-Filter

automatisches Filtern von

- Flash-Werbung
- Banner-Werbung (v3)
- Google-Werbung (v3)
- IntelliTXT-Werbung (v3)
- Layer-Werbung



# Quero Agenda

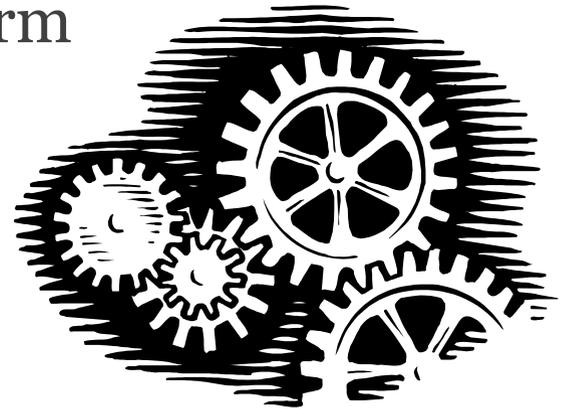
---

- Innovation Quero Toolbar  
Usability, Security, Privacy
- **Technologie**  
Windows Internet Explorer, Architektur
- Softwarequalität  
Entwicklungsmodell, Qualitätsmaßnahmen,  
Herausforderungen

# Quero **Verwendete Technologien**

---

- Microsoft Windows
- Microsoft Visual C++
- Architekturen COM und Browser Helper Object
- Frameworks ATL und WTL
- Windows Internet Explorer Platform



# Quero **Toolbar Version 2**

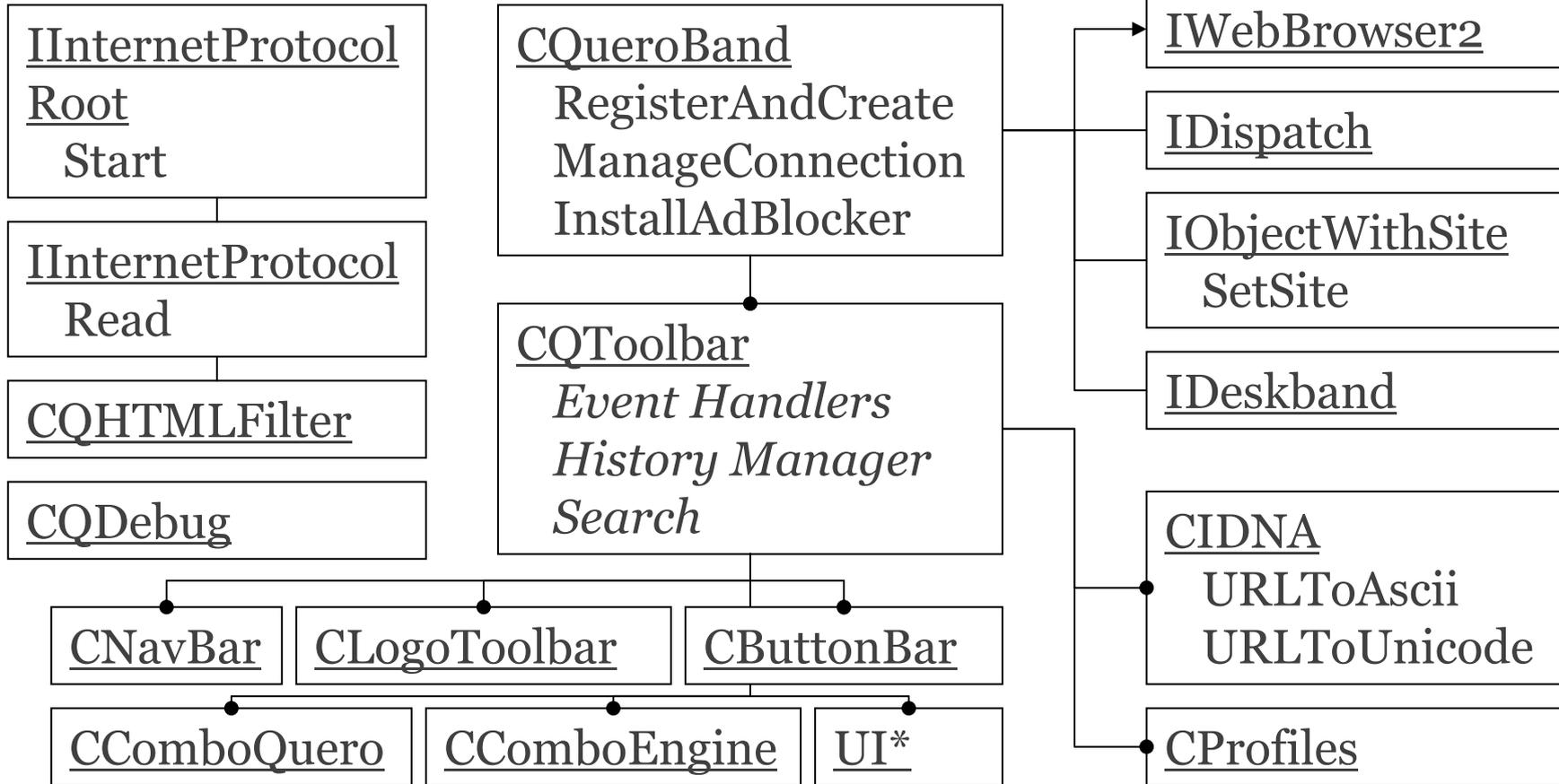
---

- OS Support: Windows 98 bis Windows 2003
- Zwei Sprachversionen: Englisch, Deutsch
- > 20.000 Lines-of-Code (LOC)
- > 10.000 Downloads zwischen 01/2005 und 01/2006
- CNET Review: 4 von 5 Sternen
- Tucows Rating: 4 von 5 Kühen

## **Version 3**

- IE7-Unterstützung
- Verbessertes Ad Blocker

# Quero Architektur



# Quero Agenda

---

- Innovation Quero Toolbar  
Usability, Security, Privacy
- Technologie  
Windows Internet Explorer, Architektur
- **Softwarequalität**  
Entwicklungsmodell, Qualitätsmaßnahmen,  
Herausforderungen

# Quero **Entwicklungsmodell**

---

Einsatz des agilen Software-Entwicklungsmodells

## **Extreme Programming (XP)**

- Kein Pflichtenheft!
- Feature-Requests während der Entwicklung (Feedback-E-Mails, Newsgroups, aktuelle Entwicklung am Markt)
- Ständige Refaktorisierung
- Kontinuierliche Architekturverbesserungen
- Kurze Release-Zyklen

# Quero Qualitätsmaßnahmen

---

- Strukturierte und objektorientierte Programmierung
- Gemeinsame Code-Base für alle angebotenen Versionen
- Source-Code-Dokumentation
- Versions-Archiv
- Detailliertes Changelog
- Bug-Verzeichnis (sowohl für Quero als auch IE)
- Feature-Request und Ideen-Sammlung

# Quero Testing

---

- Intensives Testing vor jedem Release
- Beta-Testing (geplant für Quero Toolbar 3)
- Black-/Whitebox-Testfälle für den Content-Filter
- Concurrency-Tests mit dem Quero Launcher
- Debug-Facility-Klasse (CQDebug)
  - Protokollierung von Ereignissen und Zuständen in einem Log
  - Ausgabe in der Quero Debug Console
  - Switch zwischen Release- und Debug-Build

# Quero Security

---

- Einsatz von Secure-String-Funktionen, um Buffer-Overflow-Exploits (BOE) vorzubeugen.

Beispiel:

```
TCHAR currentURL[MAXURLLENGTH];
```

```
// Unsafe
```

```
    _tcsncpy(currentURL,url);
```

```
    _tcsncpy(currentURL,url,MAXURLLENGTH);
```

```
// Safe
```

```
    StringCbCopy(currentURL,sizeof currentURL,url);
```

# Quero Testumgebung

---

- Konservative Verwendung des Heaps, um Memory-Leaks und BOE zu vermeiden.
- Testumgebung
  - 1 Entwicklungsrechner
  - 3 Testrechner mit unterschiedlichen Betriebssystemen und IE-Versionen
  - Einsatz von VM-Software

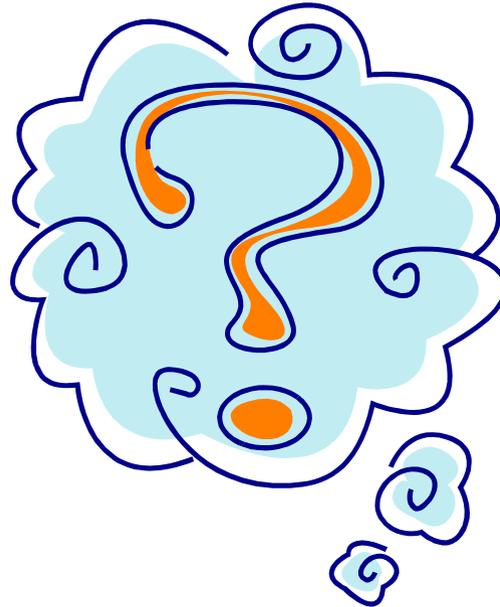
# Quero Herausforderungen

---

- Wachsende Komplexität
- Concurrency (multi-threaded, multi-process)
- Synchronisation (IPC)
- Veränderungen in der IE-Plattform (IE7)
- Große Versionsvielfalt (Windows, IE)
- Unzureichende IE-Plattform-Unterstützung (z.B. für Ad Blocking, Tabbed Browsing, UI-Anpassung)

# Quero

---



Viktor Krammer

[support@quero.at](mailto:support@quero.at)

<http://www.quero.at/>