

## Software für Webservice & System-Integration

Software Engineering, Design & Development auf IoT & IIoT-Basis

*bintellix® Software-Engineering & -Design hilft Unternehmen, die Komplexität modularer Systeme & Software in den Griff zu bekommen und schneller anpassungsfähige Produkte anzubieten.*

*Martin Richter CIO von INTERSECURITY*

## Potentiale erkennen und sinnvoll ausschöpfen

Wir verwandeln Ihre Ideen & Prozesse in **skalierbare Erfolge** mit dem Einsatz **smarter Software & intelligenter Systeme**.

Und gewährleisten so **ressourceneffiziente & perfektionierte Abläufe** in Ihrem Unternehmen, die Ihnen **langfristige Wettbewerbsvorteile** sichern.

bintellix® München ist der Spezialist für IT-Development, Software-Design, System-Integration & API-Entwicklung auf Basis von IoT- & IIoT-Technologien.

Unser Ansatz – Ihre Vorteile:

### ■ transparentes Technical Consulting

Persönliche & individuelle Beratung –  
**technologieneutral & herstellerunabhängig**



## ■ klarer Mehrwert mit IoT-Lösungen

intelligente & zukunftsfähige Systeme für **perfektionierte Prozesse**, innovative Produkte & **hervorragende Qualität**

## ■ agile Software Entwicklung

Modernste Technologien & Anwendungen für optimierte Services, Produktivität & revolutionäre Unternehmenserfolge

## Die besten Schlüsseltechnologien

### für Ihre Business-Agility & Innovationskraft

Der Schlüssel für ein **kontinuierliches Unternehmenswachstum** liegt in der **innovativen** Software-Entwicklung & agilen, **vollautomatisierten** Prozessen.

Nur so ist eine **hochflexible Business-Agility** garantiert, die von der Dynamik des Digitalen Wandels nicht getrieben wird, sondern sie **beherrscht**.

Auf diese Weise machen wir aus **zukunftsfähigen Ideen konkrete & profitable Erfolge** für Ihr Unternehmen.

### für Ihre IT Landschaft

## bintellix<sup>®</sup> – Software-Lösungen & System-Integration aus einer Hand

Unsere **Experten auf dem Gebiet Internet der Dinge** bieten Ihnen ein breites **Leistungsspektrum** rund um die **Entwicklung** und **Integration** von leistungsfähigen **Systemen & intelligenten Geräten**.

bintellix<sup>®</sup> München bietet hierzu **Software Design & Entwicklung** sowie **IT & Service Architektur**. Basierend auf **adaptiven API-Technologien**, **modularen Service Architekturen** und **zuverlässigen IoT Technologien**.

Den Schwerpunkt legen wir dabei auf 3 Bereiche:

- Software Engineering auf Basis von **Service-Integration & RESTful-API-Technologien**
- Design, Entwicklung & Test von **modularen Middle-Tier & Backend-Architekturen**
- Aufbau von **IIoT Landschaften** und **Integration von Smart Devices & digitaler Steuerungen**

Web- und Middle-Tier Systeme konzipieren wir auf **Basis von Java, C und C++** sowie den besten Standards und Frameworks zur **modularen Vernetzung**.

*Become different:  
mit dezentraler Intelligenz & Smart Clustern*

## RESTful APIs

### Agiles Schnittstellenmanagement

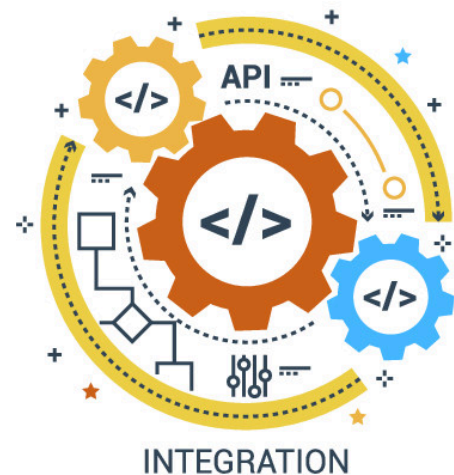
Profitieren Sie von **agilen, wartbaren & anpassungsfähigen Prozessen** über alle Geschäfts- & Produktionsebenen hinweg.

Nutzen sie die konzentrierte Kraft von API-Schnittstellen für dynamischen Workflows, flexiblen Funktionen, Online Kunden-Anbindung, zum direktem Internet-Vertrieb. Vermarkten Sie ihre Services an externe Kunden. Oder profitieren Sie einfach von der Leichtigkeit und Transparenz von API-Gateways und Service-Modulen.

### Ihr Mehrwert:

- modular und **austauschbar**
- kompakt und **integrierbar**
- maximal **standardisiert**
- **unabhängig** wartbar
- hochgradig **skalierbar**
- vollautomatisiert **testbar**

Mehr als digital vernetzt



## Web-Service & Microservices

## zukunfts offen & performance-stark

Wir bieten ein **breites Leistungsspektrum** für die Konzeption, den Aufbau und die Erweiterung von **Web-Services**.

Die Schwerpunkte liegen dabei auf **Service-Architekturen** und -Systeme in Java.

Ihr Gewinn: **schlanke, designstarke & datensichere** Kommunikations-Lösungen. Zudem vollständige **Integrationstests**, die selbst kritische Systeme non-invasiv durchlaufen können.

Als Resultat für Ihr Unternehmen:

- echte **Agilität**
- vollständige **Wartbarkeit**
- zuverlässige **IT-Security**
- zukunftsichere **Adaptivität**
- vertikale & horizontale **Skalierbarkeit**

Alles aus einer Hand: Analyse, Design, Entwicklung, Test und Support

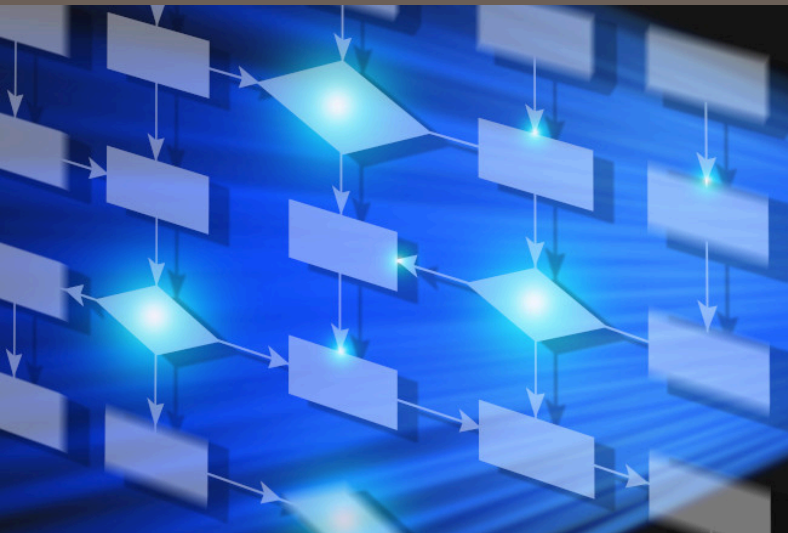
## Professionelle System-Integration & Refactoring

### schlank, sicher, skalierbar

**Erweitern Sie Ihre Unternehmenserfolge** um ein Vielfaches – durch ein breites Netzwerk aus intelligenten Systemen, Prozessen & Funktionen.

Unsere Smart Cluster Experten **bereinigen** die Digitalen Strukturen in Ihrem Unternehmen **von unnötigen Lasten & komplizierten Architekturen** mit Hilfe **hochflexibler Integrations-Layer**.Sowie Software- & Hardware-Adapter.

**Ihre Vorteile auf einen Blick:**



- 100 % **Data-Protection** & System-Performance
- steuerbare & hocheffiziente Abläufe
- **ressourcenschonende**, optimierte Prozesse
- minimales Risiko & verkürzte Time-to-Market

## Smart Devices & IIoT Steuerungen

### innovationsstark & ressourceneffizient

Zukunftsfähige IoT -& IIoT-Technologien generieren einen **unschätzbaren Mehrwert** mit langfristigen & **starken Wettbewerbsvorteilen**.

Solange Sie die **richtigen Technologien** in Ihrem Unternehmen geschickt einsetzen und nutzen.

Genau das stellen wir sicher – mit Hilfe **adaptiver & variabler Schnittstellen**, hochwertiger **Standards, Embedded Software** Modulen & flexibler Steuerungen.

Ihr Gewinn:

- ein Höchstgrad an **Konnektivität**
- überschaubare **Digitale Infrastrukturen**
- leistungsstarke Systeme



Von besonderer Bedeutung ist hierbei die intelligente Vernetzung von IIoT-Infrastrukturen, Devices & Digitalen Steuerungssystemen, um gewinnbringendes Potential aus dem Internet der Dinge zu schöpfen.

Statt auf starre, monolithische Strukturen setzen wir auf **dezentrale Intelligenz** durch integrierte Microservices. Unsere Profis im Bereich IoT System-Integration & Entwicklung prüfen & beraten Sie umfassend zu sinnvollen Systemen & Funktionen und helfen Ihnen bei der Implementierung zukunftswirksamer Technologien.

Flexibel, unabhängig & sicher.

## bintellix<sup>®</sup> – Software Engineering, System-Entwicklung & Service-Development

**Proaktiv statt reaktiv** handeln – das ist entscheidende Unterschied zwischen Erfolg & Misserfolg in einem **dynamischen Marktumfeld**, das von **schnellen Veränderungen** geprägt ist.

Mit den richtigen Technologien & **agiler Software-Entwicklung** sind Sie imstande, die Digitale Transformation **als Enabler mitzugestalten** anstatt sie „nur“ zu begleiten.

So verwandeln unsere Smart Cluster Experten zukunftsfähige Ideen **in konkrete & profitable Ergebnisse**.

Flexibel, adaptiv & intelligent.

*Connected Intelligence  
schlanke & agile Lösungen für komplexe Probleme*

### Management-Kenntnisse

Um ein technisches **Projekt erfolgreich zu gestalten**, braucht es weit mehr als IT-Kenntnisse.

**Hochwertige Management-Kompetenzen** runden das Profil professioneller Systemintegratoren nicht nur ab, sie sind ein **wesentlicher Erfolgsfaktor** für das Erreichen von Unternehmenszielen.

#### Die fünf Key-Indikatoren für ein effizientes Projektmanagement

Erfahrungsgemäß sind erfolgreiche Projekte durch 5 Punkte gekennzeichnet:

##### 1. Zieldefinition

Eine **klare Definition der Projektziele** nach dem SMART-Prinzip sorgen für Skalierbarkeit und Transparenz über die gesamte Projektdauer hinweg.

### Fachliche Kompetenz

- **Strategische Informationssystemplanung** :
  - Prozessberatung und -optimierung – *Kostensparnis durch **intelligente IT-Umsetzung***
  - Umsetzung strategischer Unternehmenskonzepte in entsprechende **IT-und ITK-Konzepte**
- **Requirements Management**:
  - Anforderungserhebung (Requirements-Engineering)
  - Umsetzung mit Risikomanagement, Änderungsmanagement, Umsetzungsmanagement
- **Software Entwicklung – Software Engineering**
- **Projekt Management – besonders *technische* Projektleitung**
- **Systemarchitektur**

## 2. Kommunikation

Regelmäßige, klare und **offene Kommunikation mit allen relevanten Fachabteilungen & beteiligten Experten** garantiert einen zuverlässigen Informationsaustausch. Missverständnisse werden so bestenfalls gleich vermieden und Engagement gefördert.

## 3. Priorisierung

Kleine, **überschaubare Einheiten**, Zwischenschritte & Teams **reduzieren die Komplexität** von Großprojekten, erleichtern die **Entscheidungsfindung** und führen zu **schnellen & greifbaren Ergebnissen**.

## 4. Technologie

**Clear Technology** hat Priorität. Die eingesetzten Technologien müssen sich **flexibel** auf neue Anforderungen **anpassen lassen** und **zukunftsfähig** sein.

## 5. Planung & Steuerung

Eine **detaillierte Projektplanung** ist notwendig für eine **risikoarme & ressourcenschonende** Projektsteuerung. Informationslücken, Abhängigkeiten, und Fehlerquellen werden dadurch **sofort erkannt & beseitigt**.

Was passiert, wenn es an entsprechenden Fähigkeiten fehlt, zeigen diverse Studien: mindestens die Hälfte der IT-Projekte erfüllen nicht die richtigen Anforderungen, ziehen sich in die Länge & fressen mehr Kosten als geplant oder werden spontan abgebrochen.

- **Konzeption**, Realisierung und Test von Individual- und Branchensoftware
- Daten Management
- Einführung und **Integration** von Software-Systemen

## Methodische Kenntnisse

- Methoden des Projektmanagement
- Prozessmanagement und IT-Strategie
- Entwicklung von Integrationsstrategien

## Projektleitung und -management

- Projektplanung und -kalkulation – *Aufwände, Termine, Ressourcen*
- Projektstrukturierung – *Projektabschnitte, Meilensteine, Arbeitspakete, Mitarbeiterzuordnung, etc.*
- **Technische Projektleitung** / Teamleitung
- Auswahl und **Steuerung von Subunternehmern** und Dienstleistern
- Erstellung von Dienst- und Werkverträgen – *Preisfindung, Gewährleistung*
- Fortschrittskontrolle und Steuerung bzgl. Terminen, Ressourcen und Risiken
- Überwachung und Steuerung von (ggf. parallel laufenden) Projektgruppen
- Erstellung regelmäßiger Reports über den Projektstand
- Koordination **interner und externer Entwickler** oder Zulieferer
- **Qualitätssicherung** von Fach- und DV-Konzepten
- Qualitätssicherung der erstellten Systemkomponenten durch **Abgleich mit den Anforderungen**
- Steuerung und Überwachung der **Abnahme** bzw. Übergabe

## Programmiersprachen

Die **Wahl der passenden Programmiersprache** ist eine knifflige Aufgabe, die ein **großes Wissen** über diverse Entwicklungsmethoden, IT-Anwendungen & Systeme voraussetzt.

Ob Java, C++ oder Python – es kommt immer darauf an, **was** Sie entwickeln lassen möchten, für **welches System** programmiert werden soll und **wozu** das Ganze letztendlich gut ist.

Während Webanwendungen noch gerne in PHP geschrieben werden, kann ohne JavaScript kein modernes User-Interface umgesetzt werden.

Manche Aufgaben sind wiederum domänenspezifisch: Zum Beispiel eignet sich R am besten für die statistische Auswertung von Daten, SQL zur Abfrage von Datenbanken. C und C++ sind die 1. Wahl in der maschinennahen Programmierung.

Daneben sind beim Software-Engineering weitere **sprachspezifische Eigenschaften** zu beachten.

Dazu zählen unter anderem die **Ausführungsgeschwindigkeit, Komplexität, Code-Qualität, Fehler-Verhalten** je Laufzeitumgebung, IT-**Security** Mechanismen & **Adaptionsfähigkeit**.

### C und C++:

Programmierung von Anwendungssystemen und Gerätetreibern für Windows, Linux and Unix. Sowie zur Embedded Software Entwicklung.

### C#:

Programmierung von **Web-Service** basierten Diensten.

### Java:

Programmierung von Client- und **Server-Systemen**.

### JavaScript / ECMAScript:

Programmierung von Client-seitigen Funktionen (Microsoft Edge, Google Chrome und Mozilla Firefox)

### Python:

Anbindung von **KI/DL** Toolsets

### XQuery:

XML Abfragesprache

## Software Methoden

Moderne Software-Entwicklung **arbeitet agil, adaptiv & sicher** nach einem ausgereiften Plan. Als **exaktes und detailliertes Vorgehensmodell** bieten sich je nach Situation verschiedene Herangehensweisen an, die auf bewährten Verfahren beruhen.

Die Vorzüge bei der Wahl der richtigen Software-Methodik: **schnellere & hochwertigere**

### Anforderungs-Management

- Fachliche Bedarfsermittlung
- Requirement-Management
- IT Konzept Erstellung
- Lastenheft / Pflichtenheft Erstellung

### Softwaremodellierung

- RESTful API Modellierung und API Design – bevorzugt im "Contract First" Approach



**Entwicklung** bei **minimalem Aufwand** an Ressourcen, Materialien & Kosten.

## Die beliebtesten Modelle in der agilen Software-Entwicklung

### ■ Scrum

Das Prinzip wird mittlerweile auch gerne im Projektmanagement genutzt. Der Clou von Scrum liegt in der **schnellen Umsetzung**, **konstanten Qualitätssicherung** und kontinuierlichen **Team-Motivation** durch die Betonung von Zwischenergebnissen.

### ■ Kanban

Kanban wurde **ursprünglich für die Fertigung & Prozess-Steuerung** in Produktionsbetrieben geschaffen (vgl. Toyota Production System). Doch auch IT-Projekte werden mit Kanban **schneller entwickelt und effizienter abgeschlossen**. Auf den Punkt gebracht: dieses Prinzip ermöglicht eine **effizientere Auslastung**, **hohe Transparenz** sowie **schnelle Iterationen** innerhalb des Projekts.

### ■ eXtreme Programmierung (XP)

XP ist eine **sehr anspruchsvolle Methode**, die auf einem **zyklischen** Vorgehen beruht. Ob Programmierung, Anforderungsmanagement oder Teamarbeit – das **agile Modeling** greift auf allen Projektebenen. Sie bietet **viel Flexibilität** und hat sich bereits häufiger bewährt.

### ■ Continuous Integration (CI)

Kontinuität ist hier das Motto. Die **kontinuierliche Integration von Software-Komponenten** vermeidet zeitintensive Rückkopplungen und Bug-Lösungen durch fehlerhafte Codes. Die CI-Methode ist eine gute Wahl, um die **Arbeitseffizienz** im Entwicklerteam und die **Qualität** des Software-Produkts zu steigern.

## DevOps (Development + IT-Operations) / DevSecOps

DevOps unterscheidet sich von den vorgestellten Software-Methoden fundamental.

- Domain Driven Design – *Deklarative Programmierung*
- Unified Modeling Language (UML) – *mit Class-, Use case-, Sequence Diagrammen*
- Entity-Relationship-Modell (ER) – *für Relationale Datenbanken*
- Software Muster – *Analyse Pattern, Design Pattern, Entwurfspattern, Architekturpattern, Antipattern*
- Objektorientierte Modellierung – *OO-Analyse, OO-Design, OO-Programmierung*

## Vorgehensmodelle

- Klassisches **Wasserfall Modell** – *strukturierte Herangehensweise*
- **Scrum** – *Agile Softwareentwicklung*
- **Kanban** – *Agile Softwareentwicklung*
- **Extreme Programming (XP)**

## Architektur-Prinzipien

- Modularisierung
- Schichten Trennung
- Spezifizierte Schnittstellen
- Kapselung von Fachlichkeit
- Wiederverwendung
- Erweiterbarkeit

## Architekturmuster

- **API-driven** consumer systems
- **Continuous Integration** – *u. a. durch Jenkins*
- **DevOps / DevSecOps** – *Vorgehensmodell zur gemeinsamen Verwendung von Tools und Frameworks durch das Software Development and IT Operation Team*
- [Privacy by Design](#)
- Security by Design
- Service-oriented architecture (**SOA**)

Im Grunde handelt es sich dabei weniger um eine Software oder Technologie als um ein **Mindset, das von Teamgeist, Lernbereitschaft & Vertrauen geprägt** ist.

Ziel ist es, von einem **beschleunigten Software-Engineering** zu profitieren und **erstklassige Produktqualität** zu gewährleisten, indem alle beteiligten Fachgebiete verstärkt zusammenarbeiten.

Damit das gelingt, müssen **sämtliche Standards intelligent vernetzt** werden. Auf diese Weise kommt es zum Beispiel zu **verkürzten Release-Zeiten** und einem **effektiveren Risikomanagement**.

- **Microservices** – *Lose gekoppelte Funktionalitäten, die per Event Bus kommunizieren*

## Software Modellierung

- Objektorientierter Entwurf:
  - OO-Analyse (**OOA**)
  - OO-Design (**OOD**)
  - OO-Programmierung (**OOP**)
- Konzeptionelles Design – **UML Design, Use-Case Diagramme**

## Qualitätssicherung

- Automatisierte Test – *vor allem Integrationstest*
- ISO9000 - ISO9004 Qualitäts-Management
- Performance und Kapazitäts-Management

## Software Standards

Fundamentales **Know-how in allen gängigen Software-Standards** ist ein Charakteristikum professioneller Entwickler & Software-Architekten.

Standards punkten einerseits mit **universellen Nutzungsmöglichkeiten, Datensicherheit, Kontrolle** und **zuverlässiger Qualität**.

Andererseits mit **unmittelbaren Vorteilen**, wie:

- verkürzte Time-to-market
- schnelle System- & Software Upgrades
- ein **hoher Automatisierungsgrad**
- sichere Updates für Hochrisiko-Releases
- unabhängige, agile Software-Entwicklung
- minimierter Kosten-Aufwand

## Hierarchische Datenstrukturen

- **Avro** Datenstruktur & Schema
- **JSON** Datenstruktur & Schema
- **XML** Datenstruktur & XML Schema / RELAX NG

## Webservices / REST APIs

- [Hypertext Application Language \(HAL\)](#) – *HAL is a simple format that gives a consistent and easy way to hyperlink between resources in your API.*
- [Hypermedia as the Engine of Application State \(HATEOAS\)](#) – *Enables dynamic usage of URL based APIs by providing context based hyperlinks*
- **Siren** – *a hypermedia specification for representing entities*
- **OpenAPI (vormals Swagger)** – **REST API Spezifikation & Publication Standard**

Heute können Unternehmen dank innovativer APIs und Micro-Services eine **eindrucksvolle Wettbewerbsstärke aufbauen**, die sich immer auf dem aktuellen Stand der Technik gründet.

- Service Discovery:
  - DNS Service Discovery (**DNS-SD**)
  - Service Location Protocol (**SLP**)
  - Universal Description Discovery and Integration (**UDDI**)
  - Web Services Dynamic Discovery (**WS-Discovery**)
- Simple Object Access Protocol (**SOAP**) – incl. *WSDL, Document Style*
- Service-Oriented Architecture Registry (SOA Registry)

## Business Prozess Modelle Business process models

- Business Process Execution Language (**BPEL**)
- Business Process Modeling Notation (**BPMN**)
- XML Process Definition Language (**XPDL**)

## Web-Server Techniken Web-Server technologies

- Enterprise Java Beans (**EJB**)
- Java **Servlets**
- Java Server Pages (**JSP**)
- **.NET** Enterprise Services
- **.NET Remoting**

## Web Standards

- HTML / XHTML
- Cascading Style Sheets (**CSS**)
- JavaScript mit **DOM**

## XML Technologien XML technologies

- XForms
- XPATH
- XQuery

■ XSL / XSLT

## Datenkommunikation Protokolle und Standards

Kennzeichen von **leistungsfähigen Systemen & Anwendungen** ist eine intelligente Kommunikation. Sie **garantiert höchste IT-Security** sowie einen **sparsamen & schnellen** Datenaustausch unter den Teilnehmern.

Wussten Sie schon, dass...

- ... ein TLS Verbindungsaufbau, wie z. B. bei https:// durchschnittlich 12x die Latenzzeit benötigt?
- ... man TLS verschlüsselte HTTP Verbindungen nicht im Cluster parallel/verteilt abarbeiten kann?
- ... man mit HTTP keine semi-permanenten Verbindungen aufbauen kann?
- ... die HTTP POST Methode keinen "Content-Range" und somit kein Recovery kennt, falls die Verbindung unterbrochen war)?
- ... TCP und UDP für semi-permanente Verbindungen regelmäßig ein Keep-Alive Signal benötigen?
- ... UDP deutlich effizienter Daten überträgt als TCP (TCP hat das Head-of-line blocking Problem, TCP braucht Bandbreite für ACK Pakete)?
- ... lokale Speicherung immer einem verteiltes Storage (egal ob iSCSI oder NFS) vorzuziehen ist?
- ... ICMP elementarer Bestandteil des IP Verbindungsaufbaus ist, aber im OSI Schichtenmodell vermeintlich eine Schicht darüber liegt?

Lückenhaftes Wissen und Halbwissen führen leicht zu mangelhaftem Einsatz von Protokollen und damit einhergehenden

## Datenkommunikation

- REST / **RESTful** – *Industrie Standard für HTTPS basierte APIs*
- **SOAP** / WSDL – *SOAP (Simple Object Access Protocol) mit dem WSDL (Web Service Description Language) Standard*
- Web Server Protokolle – *HTTP/HTTPS incl. **WebSocket***
- E-Mail Server Protokolle – ***SMTP**, POP3, IMAP, DomainKeys Identified Mail (DKIM), Sender Policy Framework (SPF), DNS-Blacklist, STARTTLS*
- Remote Shell Protokolle – *SSH, SCP, SFTP*
- Telefonie – ***SIP/SIPS**, RTP, STUN, SIP trunking, e164.arpa*

## Datei Übertragung und Speicherung

- Amazon **S3** – *HTTP basiertes File-Transfer und Speicher Protocol*
- Server Message Block (**SMB**) – *Microsoft File-Transfer und Authentication Protocol. Vormalis: Common Internet File System (**CIFS**)*
- Network File System (**NFS**) – *Unix Network File System*
- Web Distributed Authoring and Versioning (**WebDAV**) – *HTTP basiertes File-Transfer und Speicher Protocol*

## Internet Protokoll

- IP Protokoll – *IP-Protokoll **IPv4 und IPv6** (TCP, UDP, ICMP, MPTCP), IPsec, QOS/TOS, Neighbourhood Discovery*
- Dynamic Host Configuration Protocol (**DHCP**) – *für IPv4 und IPv6*
- Domain Name System (**DNS**) – *incl. Service Discovery*

Sicherheitsrisiken im Betrieb. So z. B. bei ungeprüftem Einsatz der HTTP CONNECT Methode, weil man vermeintlich nur sichere WebSocket Verbindungen nutzt.

Ein erfahrener Netzwerk Experte und System-Integrator verfügt über tiefgreifenden Wissen der zu verwendenden Protokolle. Damit kann er potentielle Probleme bereits im Vorfeld erkennen. Und diesen durch gekonnte Aufklärung und effektiven Gegenmaßnahmen sicher begegnen.

- IP Security – IP-Routing (incl. NATing), IP-Firewall, VPN mit **IPSec und WireGuard**

## Netzwerk Management

- Port-based network access control (IEEE 802.1x) – *Makes use of the physical access characteristics of IEEE 802 Local Area Networks (LAN) infrastructures in order to provide a means of authenticating and authorizing devices attached to a LAN port*
- Spanning Tree Protocol (**STP**) – *MSTP (IEEE 802.1s), RSTP (IEEE 802.1w), STP (IEEE 802.1d)*
- Virtual Local Area Network (**VLAN**) – *Ethernet VLAN Tagging*

## Security Technologien und Standards

Cyber-Sicherheit ist kein Schlagwort, sie ist **existenziell für ein digitales & wirtschaftliches Wachstum**. Doch mit dem **technologischen Fortschritt** verändern sich auch die Anforderungen an die IT Security. Auch die **Sicherheitsstandards** werden regelmäßig auf Basis neuer Technologien und veränderter Sicherheitslage nachgezogen. IT Sicherheit ist keine einmalige Anforderung, sondern eine kontinuierliche Herausforderung.

In einer sich ständig verändernden Welt, mit ständig neuen Herausforderungen, kann man nur mit einer dynamischen IT bestehen.

Diese Dynamik der Digitalisierung lässt sich auch als Chance verstehen, um eine nachhaltig sichere IT aufzubauen, welche es dem

## Datensicherheit

- **API Gateway / API Firewall** – *REST based Application Firewall*
- **BSI Richtlinien** – *IT Grundschutz Kompendium, Zertifizierung nach ISO 27001*
- Certification standards – *Kriterienkataloge: Orange Book, IT-Kriterien, ITSec-Kriterien, Common Criteria; Personal: CISM, CISSP*
- Cryptography Standards:
  - Certificate Authority (**CA**)
  - Crypto-Module (Verschlüsselung)
  - elektronische Signatur
  - Public-Key Cryptography Standards (**PKCS12**) – *z. B. für eToken*
  - Public-Key Infrastructure (**PKI**)
  - **SmartCard** – *incl. Crypto Engine*

Unternehmen ermöglicht, effektiv & langfristig wirtschaftliche Erfolge zu erzielen.

IT-Sicherheit ist damit ein wirtschaftlicher Faktor von enormen Ausmaß, der von IoT-Spezialisten ein breites Spektrum an Wissen und Erfahrung fordert:

- die Planung neuer Sicherheitskonzepte
- die schnelle Integration von innovativen Funktionen & Systemen
- die kontinuierliche Pflege der herkömmlichen IT-Security
- Emergency & Recovery Scenario Planung

Sogenannte Pentests, 2-Faktor-Authentifizierungen, Token-Access und Single-Sign-On Verfahren gehören zum **Standard-Repertoire** eines IT Security Verantwortlichen.

Ein einschlägiger Experte versteht sich darüber hinaus auf den gezielten Einsatz von fortgeschrittenen Sicherheitstechnologien wie Honeypots bzw. Honeynets und Intrusion Detection Systems (IDS).

- Authentication / Authorization:
  - JSON Web Token (**JWT**) – *open standard (RFC 7519) that defines a JSON object for secure access*
  - **Kerberos** – *Token*
  - Microsoft **Active Directory** (AD) – *LDAP like Domain Administration*
  - **Open LDAP** – *Open Source LDAP Benutzerverzeichnis*
  - Simple and Protected GSSAPI Negotiation Mechanism (**SPNEGO**) – *xxx*
- Firewall:
  - diverse Hardware Firewalls
  - IP-Tables
  - Linux IP Filter
  - Sandbox Systeme
- **Honeypot**
- Network based Intrusion-Detection-System (NIDS)
- Host Based Intrusion-Detection-System (HIDS)
- Proxy – *Security Policy Rules, Konfiguration unterschiedliche Proxy Server*
- **Sandboy**
- Single-Sign-On (**SSO**) – *diverse Systeme wie z. B. CA SiteMinder*
- VPN – *diverse Open Source VPN Implementationen*

## Datenschutz

Datenschutz-Grundverordnung (**DSGVO**) / GDPR

## Consumer Internet of Things (CIoT) Technik

## Bussysteme zur Gebäudeautomatisierung

- [Constrained Application Protocol \(CoAP\)](#) – *ein von der Internet Engineering Task Force*

Das Internet der Dinge bietet Verbrauchern ein **ungeahntes Spektrum an Möglichkeiten**.

Unternehmen sollten jedoch auch die **Gefahren beachten**, die in der Vielfalt an Optionen, Methoden & Technologien lauern.

Durch den **Mangel an einheitlichen Standards im CloT**, braucht jeder Hersteller ein eigenes Verfahren zur Implementierung, Konfiguration, (Firmware-)Aktualisierung, Überwachung etc. seiner Produkte.

Um die verschiedenen Devices trotzdem irgendwie miteinander zu verbinden, müssen meist **proprietäre Cloud-Lösungen** erhalten. Eine vermeintliche Lösung, die **mehr Probleme** mit sich bringt als sie beseitigt. Auch bei der Firmware-Aktualisierung ergeben sich einige Schwierigkeiten.

Das alles lässt sich vermeiden, wenn sich ein guter CloT-Architekt auf **folgende Bereiche konzentriert**:

- zentrales Konfigurationsdaten-Management
- zentrale Steuerung & Kontrolle der Kommunikationspfade über alle Devices hinweg
- einheitliche Kommunikationsschicht
- dezentrale und redundant ausgelegte Steuerung, die alle Devices zu einem intelligenten Netzwerk verknüpft

Kurzum: Connected Intelligence

*(IETF) entwickeltes Web-Transfer-Protokoll mit parallelen zu REST*

- [DALI](#) – Lichtsteuerung
- [KNX sowie KNX/IP](#) – xxx
- Standard Motor Interface (SMI)
- [M-Bus](#)
- Zigbee – unter anderem Philips Hue und Zigbee Light Link (ZLL)

## Home Automation-Software

- [OpenHAB](#) – highly adaptable control, written in Java
- [ioBroker](#) – flexible and modular application for the IoT and Smarthome written in Javascript

## Microcontroller / Prototypen Hardware

- [Arduino](#) – Uno, Leonardo, Nano Every
- [Espressif Systems](#) – ESP32, ESP8266
- [Raspberry PI](#) – Uno, Leonardo, Nano Every

## Industrial Internet of Things (IIoT) Technik

Die Industrie 4.0 ist auf die Anwendung des Internets der Dinge im **produzierenden** und **industriellen** Umfeld spezialisiert.

## Feldbus und Messaging

- CAN-Bus Adapter – xxx
- Direct Number Control (DNC) – *Ist keine Bus sondern ein serialer Adapter Standard*
- [Modbus](#) – *de-facto industrial standard for serial communication*

Dank Künstlicher Intelligenz, Machine Learning & modularen Systemen ergeben sich für **Herstellung & Industrie immense Vorteile**, die sich **dauerhaft** nutzen lassen.

Zum Beispiel kann die Produktion & der Transport von Waren im Betrieb deutlich an Effizienz gesteigert werden. Auch die Wartung von Anlagen & Geräten läuft automatisch, hocheffektiv und vorausschauend ab.

Darüber hinaus ermöglicht das IIoT die **Erschließung neuer Geschäftsfelder**, **Prozessoptimierungen in Echtzeit** und eine **verbesserte Sicherheit** – sowohl für Mitarbeiter als auch Maschinen.

Das ist jedoch längst nicht alles: Mit einer leistungsstarken ITK-Architektur lassen sich verschiedenste **Herausforderungen** wie sichere Kommunikation, dezentrale Vernetzung oder die Verwaltung von Zugriffsauthorisierungen **zuverlässig meistern**.

Große Anforderungen muss auch die ITK-Infrastruktur erfüllen, um Echtzeit-Steuerung, Testsimulationen, kundenspezifische Workflows und präemptives Management zu gewährleisten.

- [MQTT](#) – *machine-to-machine (M2M) connectivity protocol*

## Interexchange

- [Open Platform Communications United Architecture \(OPC UA\)](#) – *Datenaustauschstandard für die industrielle Kommunikation (Maschine-zu-Maschine- oder PC-zu-Maschine-Kommunikation)*
- [universal machine technology interface \(umati\)](#) – *Adaption von OPC UA und Marke der Maschinenbauverbände VDMA und VDW*
- [Reference Architecture Model for Industry 4.0 \(RAMI 4.0\)](#) – *IEC-62541-Standard OPC Unified Architecture (OPC UA) konforme Musterarchitektur für Kommunikation*

## Platform Software

- [Coaty](#) – *The lightweight open-source framework for Collaborative IoT*

## Agiles Software Engineering & IT-Development für B2B, B2C & B2E

### Wachsen durch Innovation, überzeugen mit Einzigartigkeit

Als **Full Service-Agentur** mit einem Team aus **versierten** Software-Ingenieuren, Systemintegratoren und erfahrenen Anwendungsentwicklern **optimieren wir Vertriebs-, Service- und Produktions-Prozesse** nach Ihrem Bedarf.

So können Sie sich auf Ihre **Kernkompetenzen und Kunden konzentrieren**, während wir Ihr Unternehmen **nach Ihren Vorstellungen** weiterentwickeln.

### Weitere Anwendungsfelder für Connected Intelligenz:



#### Unternehmen

📍 bintellix GmbH & Co. KG  
Geigenbergerstr. 7a  
81477 München  
Deutschland

#### Comunity

📘 facebook.com/bintellix  
🐦 twitter.com/bintellix  
🐙 github.com/twitter

#### Kontakt

☎ +49 89-7507504-0  
📠 +49 89-7507504-99  
✉ info@bintellix.com  
📄 Kontaktformular

#### Unternehmensgruppe



**INTERSECURITY**