

rTR Diskussionspapier, 01.2018

## SOFTWAREOBSOLESZENZ ALS HERAUSFORDERUNG FÜR DIE REPARATUR

Die Lebens- und Nutzungsdauer von Software entscheidet immer öfter darüber, wie lange Produkte genutzt werden können, wie hoch die Potenziale für die Weiter- und Wiederverwendung sind und ob eine Reparatur möglich oder sinnvoll ist. Dieses Papier diskutiert Ursachen und Wirkungen von Softwareobsoleszenz auf den Produktlebenszyklus und zeigt Lösungsansätze, um negative Auswirkungen zu verhindern oder abzumildern.

Als eine Folge der zunehmenden Digitalisierung hält Software heute und in Zukunft immer weiter Einzug in die Produktwelt: Elektromobilität, Smartphones, Smart Home Produkte wie internetfähige Thermostate, Beleuchtungen oder Küchenherde, Internet der Dinge (IoT) und Industrie 4.0 sind heute alle maßgeblich auf Software angewiesen. Auch immer mehr Unternehmen konzentrieren sich auf Software- und IT-Dienstleistungen – BITKOM berichtet, dass die Umsätze mit IT-Dienstleistungen, inklusive Projektgeschäft und IT-Beratung ein überdurchschnittliches Wachstum verzeichnen und 2017 auf 39,0 Milliarden steigen.<sup>1</sup> Die schnelle Weiterentwicklung und Verbreitung von Software bietet jedoch nicht nur neue Nutzungs- und auch Einsparpotenziale, sondern verursacht auch unerwünschte ökologische und soziale Auswirkungen.

Die vorzeitige Obsoleszenz von Produkten und der damit verbundene Ressourcenverbrauch können nur dann verhindert werden, wenn die herstellenden Unternehmen freiwillig oder durch Regelungen der deutschen oder EU-Politik die Produktentwicklung und die Rahmenbedingungen der Produktnutzung verbessern. Aber auch die Verbraucherschutzpolitik muss dieses Thema rechtzeitig auf die Agenda setzen und nachhaltig lösen.

---

<sup>1</sup> BITKOM (2017): Bitkom-Branche schafft in diesem Jahr 21.000 neue Jobs, unter: [https://www.bitkom-rese-arch.de/epages/63742557.sf/de\\_DE/?ObjectPath=/Shops/63742557/Categories/Presse/Pressearchiv\\_2017/BitkomBranche\\_schafft\\_in\\_diesem\\_Jahr\\_21000\\_neue\\_Jobs](https://www.bitkom-rese-arch.de/epages/63742557.sf/de_DE/?ObjectPath=/Shops/63742557/Categories/Presse/Pressearchiv_2017/BitkomBranche_schafft_in_diesem_Jahr_21000_neue_Jobs) (02.08.2017)

## **SOFTWAREOBSOLESZENZ BESCHRÄNKT DIE NUTZUNGSDAUER VON PRODUKTEN**

Die Softwareabhängigkeit bewirkt bei vielen Produkten einen vorzeitigen Funktionsverlust (funktionelle Obsoleszenz), indem die technische Nutzbarkeit der Software verloren geht oder der subjektive Nutzen des Gerätes aus Sicht des Anwenders nicht mehr gegeben ist:

- Das Smartphone erhält ein Update oder Upgrade auf die neueste Version des Betriebssystems. Durch die stärkere Beanspruchung der Hardwarekapazitäten wird das Smartphone in der Benutzung jedoch für den/die NutzerIn unerträglich langsam. Ist das Upgrade erforderlich, um das Gerät weitzernutzen zu können, und/oder nach der Installation kein Downgrade auf vorherige Versionen möglich bleibt dem/r NutzerIn keine andere Wahl, als die Hardware zu wechseln;
- Der überwiegende Teil von aktuellen Betriebssystemen und Softwareprogrammen erfordert mit jeder neuen Produktversion (-generation) den Einsatz von neuer leistungsfähigerer Hardware. In diesen Fällen sind IT-NutzerInnen dazu gezwungen, sich neue PCs und Hardware anzuschaffen, wenn sie nicht einen mit Kosten und Aufwand verbundenen Wechsel von Betriebssystem oder Softwareprogramm in Kauf nehmen wollen;
- Kündigt der Hersteller den Software-Support für ein Gerät auf, zwingt er NutzerInnen zum Upgrade oder zum Wechsel auf ein neues System. Gibt es keine Upgrade-Möglichkeit und auch keine Möglichkeit zum Wechsel auf ein neues System mit bestehender Hardware, bleibt dem/der NutzerIn nur ein Wechsel der Hardware;
- Peripheriegeräte können nicht mehr genutzt werden, da die Hersteller keine aktualisierten Softwaretreiber bei Upgrades der Gerätesoftware mehr zur Verfügung stellen.

## **SOFTWARE ALS HINDERNIS FÜR DIE REPARATUR UND REUSE**

Vor allem zwei Aspekte beeinträchtigen die Reparatur von Geräten: Sowohl die fehlende Bereitstellung von Software zur Diagnose von Problemen wie auch geschlossene Software, die keine Diagnose durch eigenständige Reparatursoftware ermöglicht:

- Geschlossene und geschützte Systeme (proprietäre) verhindern eine herstellereunabhängige Reparatur oder Wartung durch die NutzerInnen oder Drittparteien. Ein(e) möglicher Weiterbetrieb und -entwicklung der Systeme durch eine (Open Source) Community ist ebenfalls ausgeschlossen;

- Der Reuse-Faktor von IT-Produkten sinkt erheblich, indem Dritten der Zugang zu herstellerspezifischen Diagnosetools, Software-Frameworks, Dokumentationen und dem Quellcode der Software erschwert wird.

## **DIE FOLGEN FÜR VERBRAUCHERINNEN UND UMWELT**

Die erhöhte Softwareabhängigkeit geht in den meisten Fällen mit spürbaren Nachteilen für die VerbraucherInnen einher und erhöht gleichzeitig die sozio-ökologischen Kosten durch die Neuanschaffung:

- Funktionierende Geräte und Technik werden durch Software entwertet und unbrauchbar gemacht, obwohl die Hardware noch funktionstüchtig ist;
- Studien belegen hinreichend, dass eine verlängerte Lebensdauer von IT-Produkten gegenüber der Neuanschaffung mit einem geringeren Ressourcenaufwand und weniger Umweltauswirkungen einhergehen. Eine vorzeitige Softwareobsoleszenz wirkt dem Ziel der Nachhaltigkeit entgegen.

In der Entschließung zum Thema „Längere Lebensdauer für Produkte: Vorteile für Verbraucher und Unternehmen“<sup>2</sup> weist das Europäische Parlament außerdem darauf hin, dass ein Großteil des elektronischen Abfalls entsteht, weil Hersteller keine Softwareaktualisierungen zur Verfügung stellen, die mit der Hardware kompatibel sind.

Das heute bereits bestehende Machtgefälle zwischen NutzerInnen und Herstellern und zwischen Herstellern und TechnikerInnen und unabhängigen Reparaturwerkstätten wird durch die Software-Barriere ebenfalls weiter verstärkt. Hersteller stellen bereits jetzt Ersatzteile, Reparaturinformationen und Diagnosetools oft nur ihren Vertriebswerkstätten zur Verfügung oder verlangen hohe Preise für die Nutzung durch Dritte. Unabhängige Werkstätten können deswegen nicht mehr reparieren oder eine Reparatur nur zu hohen Preisen anbieten. Es gibt dadurch weniger Wettbewerb auf dem Reparaturmarkt und für VerbraucherInnen weniger Auswahl bei der Suche nach einer guten, wohnortnah gelegenen Werkstatt.

---

<sup>2</sup> Europäisches Parlament (2017): Längere Lebensdauer für Produkte: Vorteile für Verbraucher und Unternehmen, unter: <http://www.europarl.europa.eu/sides/getDoc.do?type=REPORT&reference=A8-2017-0214&language=DE>

## LÖSUNGSANSÄTZE

Softwareprodukte müssen als ein Teil der integrierten Produktpolitik (IPP)<sup>3</sup> der Europäischen Kommission behandelt werden, deren Ziel es ist die Umweltauswirkungen eines Produktes über den kompletten Lebenszyklus zu minimieren. Die folgenden Lösungsansätze stellen Vorschläge für eine Verbesserung der Umweltleistung von Softwareprodukten dar und erhöhen gleichzeitig den Verbraucherschutz.

### 1.) Gestaltung von Software gegen vorzeitige Obsoleszenz

Software sollte zu jedem Zeitpunkt nur die Hardwareressourcen (Rechenleistung, Speicherplatz, Bandbreite im Netzwerk) beanspruchen, die für die jeweilige Aufgabe benötigt werden. Weniger leistungsfähige, weil ältere Hardware, könnte länger genutzt werden, wenn von neuen Softwareversionen nur die wirklich benötigten Module installiert und betrieben würden.<sup>4</sup>

Das bedeutet:

- Bedarfsgerechte Skalierung und Nutzbarkeit der Hardware- und Softwarekapazitäten, z.B. Nutzern die Möglichkeit geben, z.B. bei Bildern, Spielen und Videos mit niedriger Auflösung zu arbeiten bzw. skalieren zu können;
- Unterstützung sparsamer Datenformate in Hinblick auf Bandbreite und Speicherplatz;
- Modular entwickelte Software, um eine Weiternutzung der Software bei neuen Generationen der Hardware zu ermöglichen. Software sollte von der Hardware soweit wie möglich abstrahiert werden;
- „Sustainability Reviews and Previews“ während der Software-Entwicklungsphase.

Kompatibilität zwischen Formaten, Software und Geräten gibt die Möglichkeit, Software weiterzuverwenden, wenn sich die Umstände ändern, unter der sie verwendet wird.

- Unterstützung offener Standards bei Datenformaten;
- Flexibilität bezüglich verwendbarer Peripheriegeräte, um Beschaffungszwänge zu minimieren.

---

<sup>3</sup> Siehe: EU-Kommission: Grünbuch zur integrierten Produktpolitik. Brüssel, den 07.02.2001 KOM (2001) 68 endgültig final.

<sup>4</sup> Siehe zum Beispiel: UBA-finanziertes Forschungsvorhaben „Sustainable Software Design“.

Die Bedeutung von Green-IT und Green by IT bzw. Software und ihr Einfluss auf Nachhaltigkeit wird an verschiedenen Stellen bereits betrachtet.<sup>5</sup> In der Entwicklungsphase der Software anzusetzen ergibt damit Sinn. Der Einfluss von Software auf Obsoleszenz und auf Reparatur kann hier mitgedacht werden.

## 2.) Bereitstellung von Software

Ältere, aber noch funktionstüchtige Geräte sollten mit älterer Software weiter betrieben werden können. Um das zu erreichen, sind folgende Maßnahmen erforderlich:

- Software wird angemessen aktualisiert; Updates und Upgrades sind verfügbar;
- Weiterverfügbarkeit von Software auch für ältere Geräte und Peripheriegeräte;
- Möglichkeit, Software-Upgrades und Updates rückgängig zu machen, wenn sich herausstellt, dass sie noch funktionsfähige Geräte durch höhere Anforderungen lahmlegen.

Sollte durch den Hersteller keine Verfügbarkeit der Software mehr gegeben sein, muss dafür Sorge getragen werden, dass eine andere Möglichkeit für die VerbraucherInnen besteht. Hier können Open Source Initiativen einspringen, doch auch diese benötigen für die Entwicklung von freier Software entsprechende Voraussetzungen:

- Software Designpatterns und Architekturen, die eine lange Nutzbarkeit gewährleisten, wie plattformunabhängige Entwicklungskonzepte statt nativer Softwareanwendungen;
- Zertifizierungen von Drittanbieter-Treibern, damit diese gleichberechtigt mit herstellereigenen Treibern verwendet werden können;
- Offenlegung des Quellcodes bei Einstellung des Supports durch den Hersteller.

---

<sup>5</sup> Zum Beispiel: Übersicht bei Naumann, Stefan; et. al. (2015): Sustainable Software Engineering: Process and Quality Models, Life Cycle, and Social Aspects; in: L.M. Hilty and B. Aebischer (eds.): ICT Innovations for Sustainability: Advances in Intelligent Systems and Computing 310, S. 191-205. Gröger, Jens; et. al. (2015): Nachhaltige Software: Dokumentation des Fachgesprächs „Nachhaltige Software“ am 28.11.2014 beim Umweltbundesamt, unter: <https://www.oeko.de/oekodoc/2318/2015-489-de.pdf> (02.08.2017)

### 3.) Kompatibilität von Upgrades und Updates

Damit die noch funktionierende Hardware nicht wegen obsoleter Software entsorgt werden muss, müssen Upgrades und Updates auf Kompatibilität zu einander und zu älteren Geräten hin gestaltet sein. Das bedeutet:

- Die Hersteller müssen die NutzerInnen von der Updatepflicht befreien;
- Gewährleistung der Abwärtskompatibilität, sodass ein Datenaustausch zwischen neueren und älteren Versionen der Software möglich ist und die Funktion älterer Software weiterhin gewährleistet ist;
- Updates bei verwendeten Peripheriegeräten und den Geräten selbst müssen kompatibel sein, damit eine Vernetzung möglich bleibt.

### 4.) Reparierbarkeit

- Bereitstellung von offenen Programmierschnittstellen (Kurz API - application programming interface) für die Anbindung externer Programme an die vorhandene Software;
- Einsatz von Open Source oder Open Innovation Lösungen zur Förderung der Herstellerunabhängigkeit und Nutzerautonomie;
- Alle für die Reparatur benötigte Software steht den Reparateuren zur Verfügung. Eine Fehlerdiagnose muss auch am Ort selbst möglich sein, nicht nur durch Werkstechniker und nicht nur über Fernwartung;
- Es braucht eine Gegenüberstellung aller technischen Lösungen, Sicherheitsvorkehrungen und Softwarelösungen, die zu einer Einschränkung hinsichtlich der freien Reparierbarkeit führen.

## 5.) Kennzeichnung und Informationen für VerbraucherInnen

Um die VerbraucherInnen auch hinsichtlich der Software-Nutzung transparenter zu informieren, sind folgende Schritte erforderlich:

- Klare Benennung und Kennzeichnung von geplanten Software Lifetimes in den AGBs und bei Abschluss von Lizenzverträgen. Benennung des Mindestzeitraums, in dem für Betriebssysteme Sicherheitsaktualisierungen bereitgestellt werden;
- Darstellung der Auswirkung von Software-Aktualisierungen auf den Betrieb des Geräts und welche Bedeutung sie für die Sicherheit des Gerätes haben;
- Informationen, inwiefern Software-Aktualisierungen und andere Aktualisierungen mit den eingebetteten Betriebssystemen, die den VerbraucherInnen zur Verfügung gestellt werden, kompatibel sind;
- Informationen darüber, welche Möglichkeiten zur Aufrüstung der Software es gibt.
- „Awareness Building“ für die Problematik Softwareobsoleszenz in der Öffentlichkeit und in Fachkreisen.

## 6.) Politische Maßnahmen für Standardisierung, Zusammenarbeit und Kommunikation

Gerade wenn Software und Hardware getrennt voneinander entwickelt werden, oft in verschiedenen Ländern, ist ein übergreifender Rahmen gefragt, der Unternehmen und Initiativen gegen Obsoleszenz die Möglichkeit bietet, zusammenzuarbeiten. Dazu braucht es:

- Europäische Binnenmarkt-Strategie gegen Obsoleszenz bei Soft- und Hardware;
- Zusammenarbeit von Unternehmen und Initiativen auf einzelstaatlicher Ebene;
- EU-weit geltende Definition des Begriffs „geplante Obsoleszenz“ für materielle Güter und Software;

Einrichtung eines unabhängigen Testsystems zu Prüfung vorzeitiger Obsoleszenz bei Produkten;

- Förderung freier Hard- und Software-Initiativen.



## ARBEITSKREIS SOFTWARE OBSOLESENZ DES RUNDEN TISCHES REPARATUR

Der Großteil der Produktwelt unterliegt heute bereits Regulierungen wie der CE-Kennzeichnungspflicht oder Ökodesignrichtlinie für energieverbrauchsrelevante Produkte. Die Regulierung von Software, um deren positiven Einfluss auf den Nutzwert und die Nutzungsdauer von Produkten zu erhöhen, steht noch aus.

Auch gibt es Überschneidungen zwischen Verbraucherprodukten und Geräten der Industrie, beispielsweise wenn es um Softwareobsoleszenz und Peripheriegeräte, gerade auch Anlagen in der Produktion, geht. Daraus ergeben sich möglicherweise nützliche Synergien.

Der Runde Tisch Reparatur widmet sich im Rahmen seiner Zusammenarbeit im Arbeitskreis „Softwareobsoleszenz“ den Ursachen, Wirkungen und Potenzialen zur Verbesserung der Umweltsleistung und Erhöhung des Verbraucherschutzes bei softwarerelevanten Produkten und Dienstleistungen. Interessierte Fachkreise sind aufgerufen, mit dem Arbeitskreis in Kontakt zu treten.

### Kontakt

Christine Ax

Runder Tisch Reparatur

m: [christine.ax@runder-tisch-reparatur.de](mailto:christine.ax@runder-tisch-reparatur.de)

W: [www.runder-tisch-reparatur.de](http://www.runder-tisch-reparatur.de)

### Unter Mitarbeit von

Christine Ax – rTR

Antonia Reichwein – Germanwatch e.V./rTR

Erik Poppe – SUSTAINUM-Institut für

zukunftsfähiges Wirtschaften Berlin eG

### Redaktion

Cornelia Heydenreich – Germanwatch e.V.

### Mit freundlicher Unterstützung von

Gefördert durch:



Bundesministerium  
für Umwelt, Naturschutz,  
Bau und Reaktorsicherheit

aufgrund eines Beschlusses  
des Deutschen Bundestages



Die Verantwortung für den Inhalt liegt bei den Autorinnen und Autoren.