

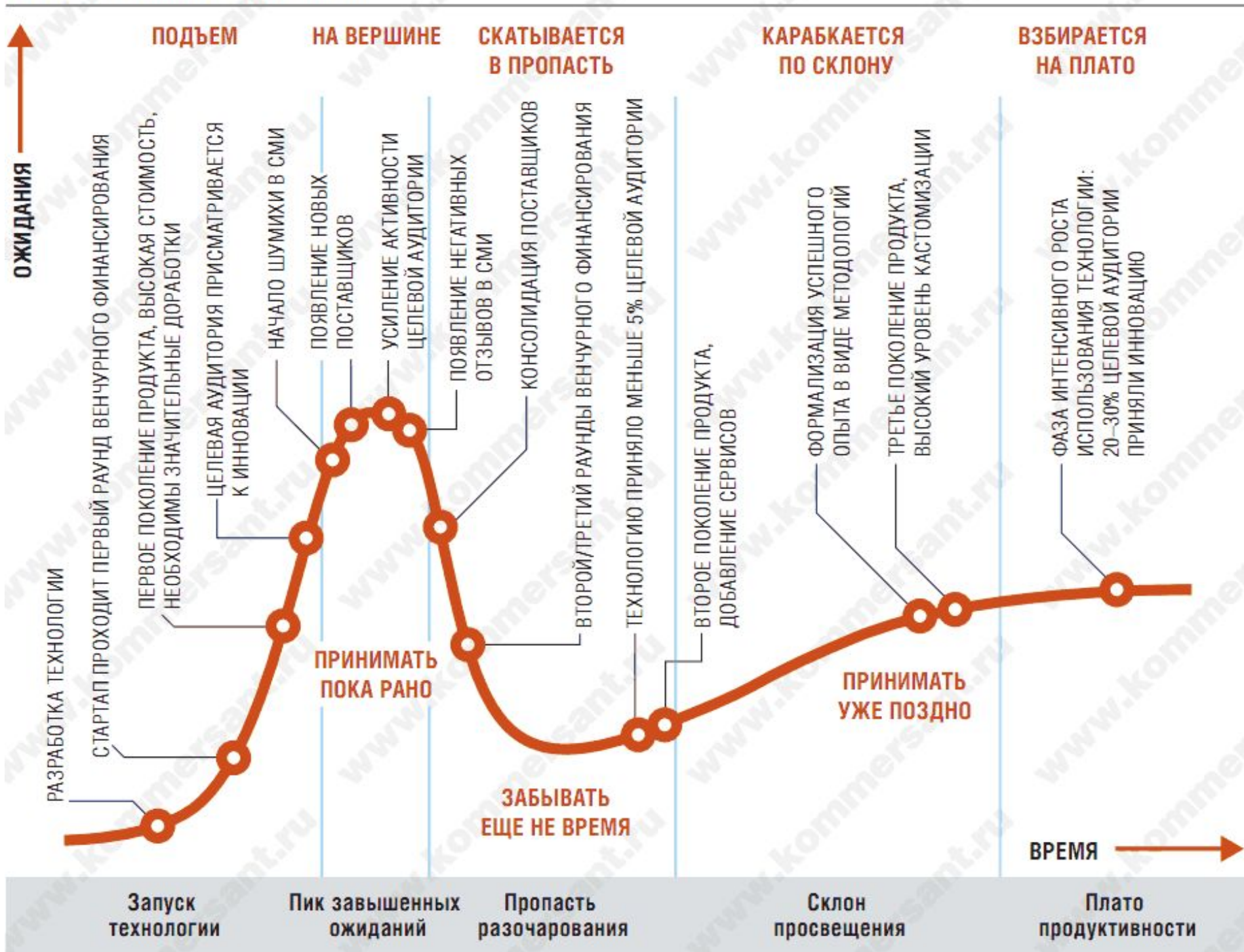
Цикл зрелости технологий

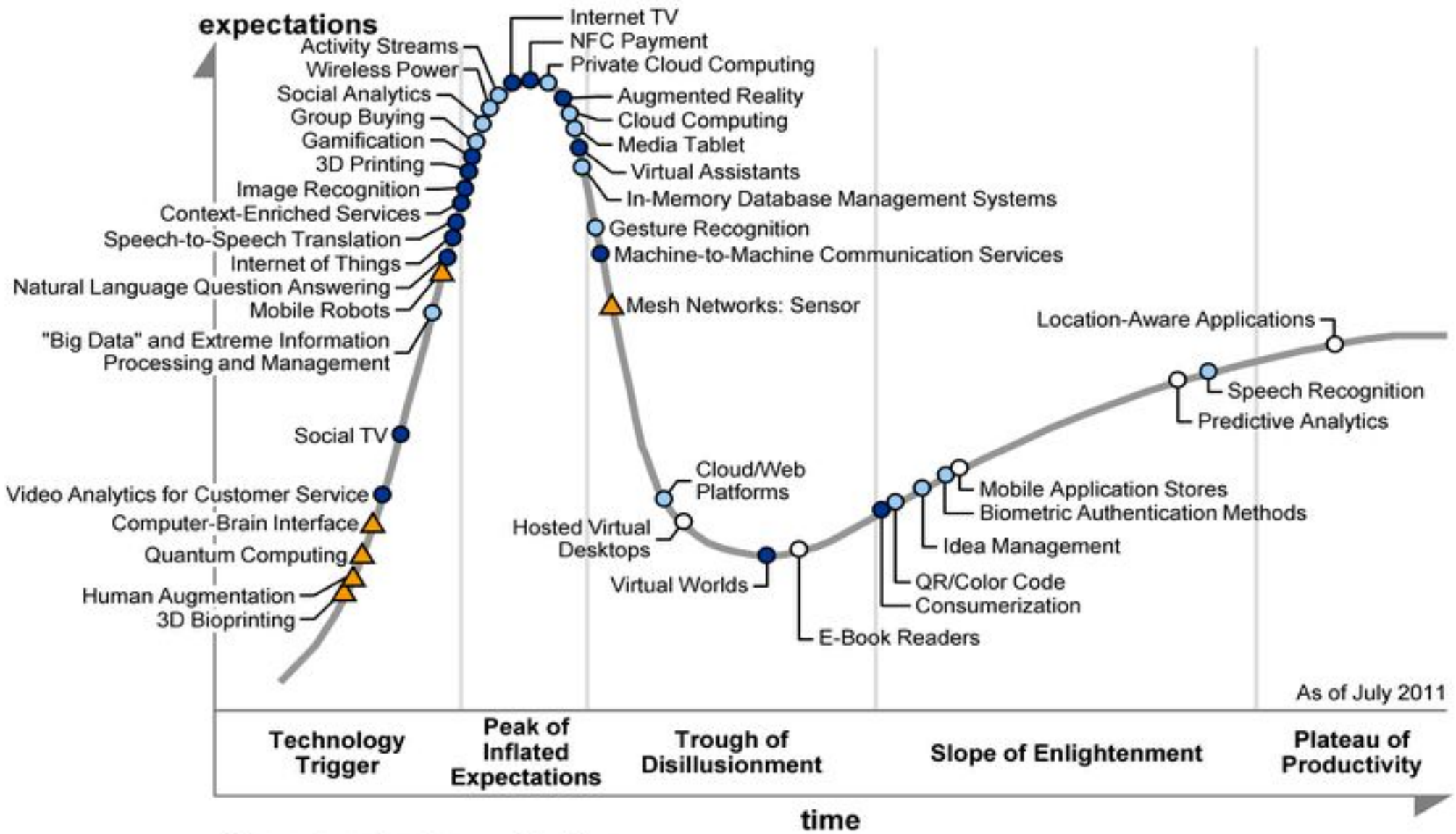
В 1995 году Gartner ввела в употребление понятие «цикл зрелости технологий» (англ. Hype cycle, также упоминается как цикл ажиотажа, шумихи, общественного интереса к технологиям, ожиданий от технологии, признания технологии). Понятие широко используется для прогнозирования и объяснения тех или иных тенденций, связанных с появлением какой-либо новой технологии. Суть понятия такова: каждая технологическая инновация в процессе достижения зрелости проходит несколько этапов, каждый из которых характеризуется различной степенью интереса со стороны общества и специалистов.

1. **технологический триггер** (англ. technology trigger) — появление инновации, первые публикации о новой технологии;
2. **пик чрезмерных ожиданий** (англ. Peak of Inflated Expectation) — от новой технологии ожидают революционных свойств, технология, благодаря новизне, становится популярной и предметом широкого обсуждения в сообществе;
3. **избавление от иллюзий** (англ. Trough of Disillusionment) — выявляются недостатки технологии, а потеря новизны не способствует восторженным публикациям, в сообществе отмечается разочарование новой технологией;
4. **преодоление недостатков** (англ. Slope of Enlightenment) — устраняются основные недостатки, интерес к технологии медленно возвращается, технология начинает внедряться в коммерческих проектах;
5. **плато продуктивности** (англ. Plateau of Productivity) — наступление зрелости технологии, сообщество воспринимает технологию как данность, осознавая её



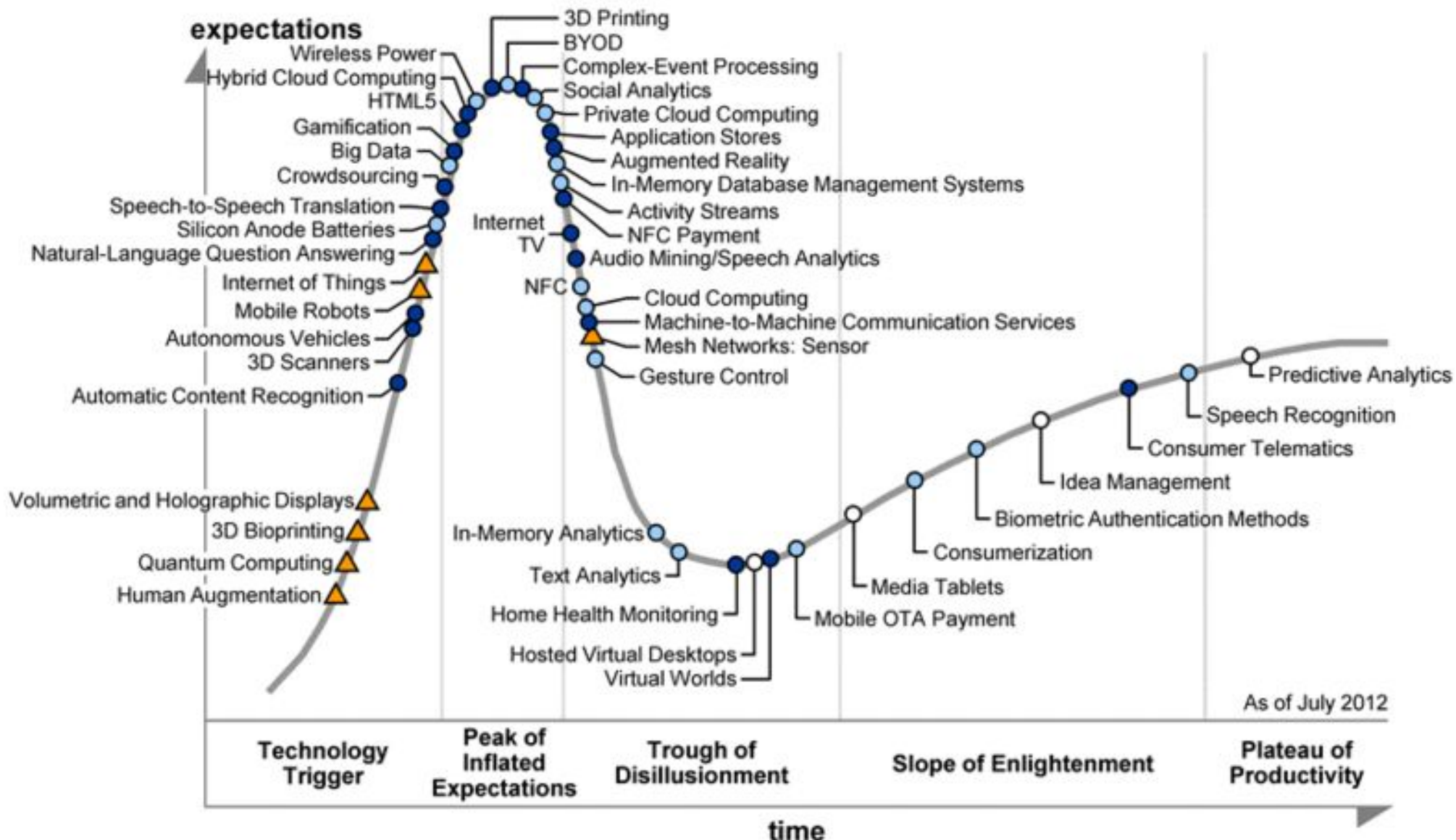
ЦИКЛ ЗРЕЛОСТИ ТЕХНОЛОГИЙ





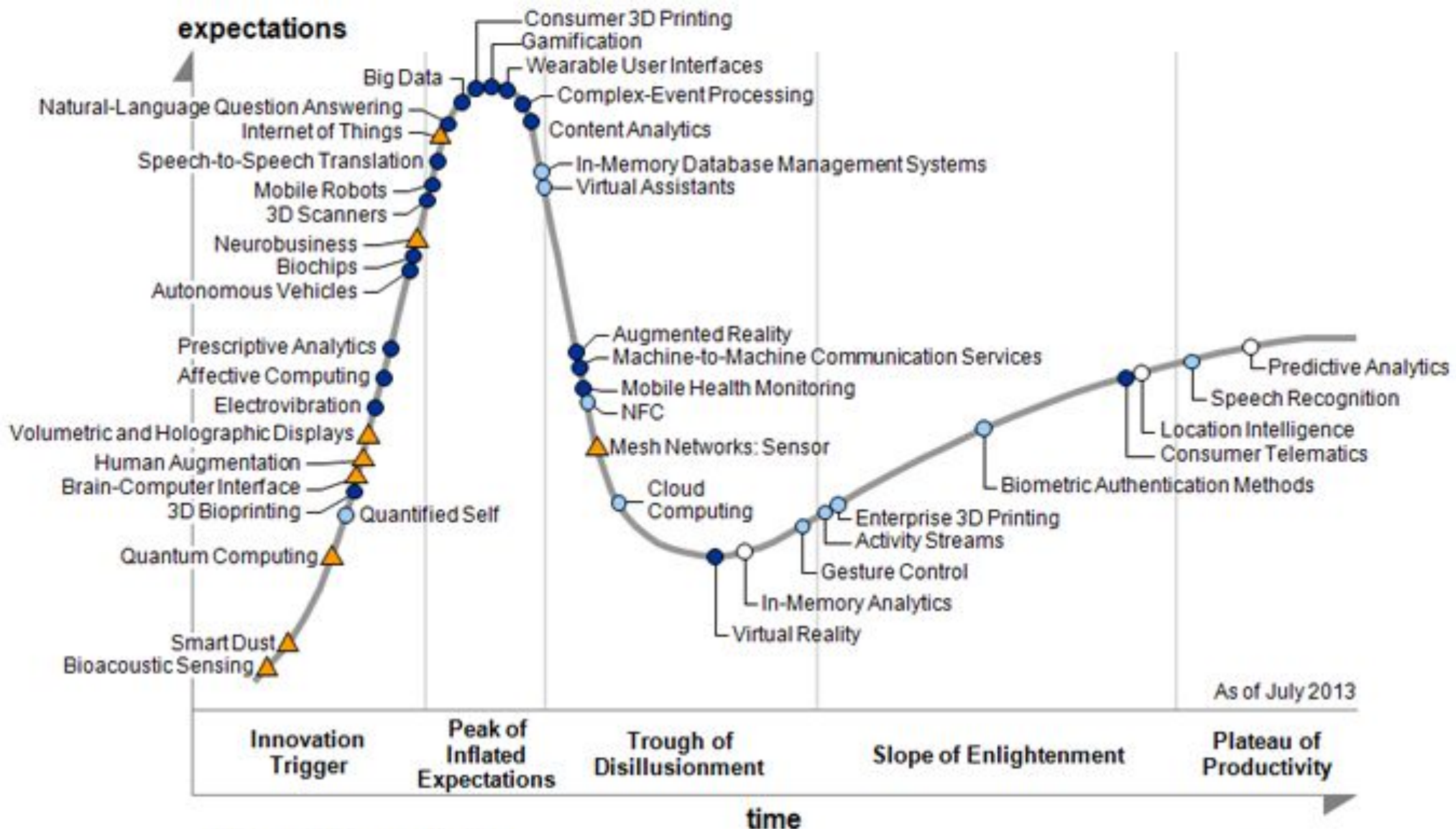
Years to mainstream adoption:

- less than 2 years
- 2 to 5 years
- 5 to 10 years
- ▲ more than 10 years
- ⊗ obsolete before plateau



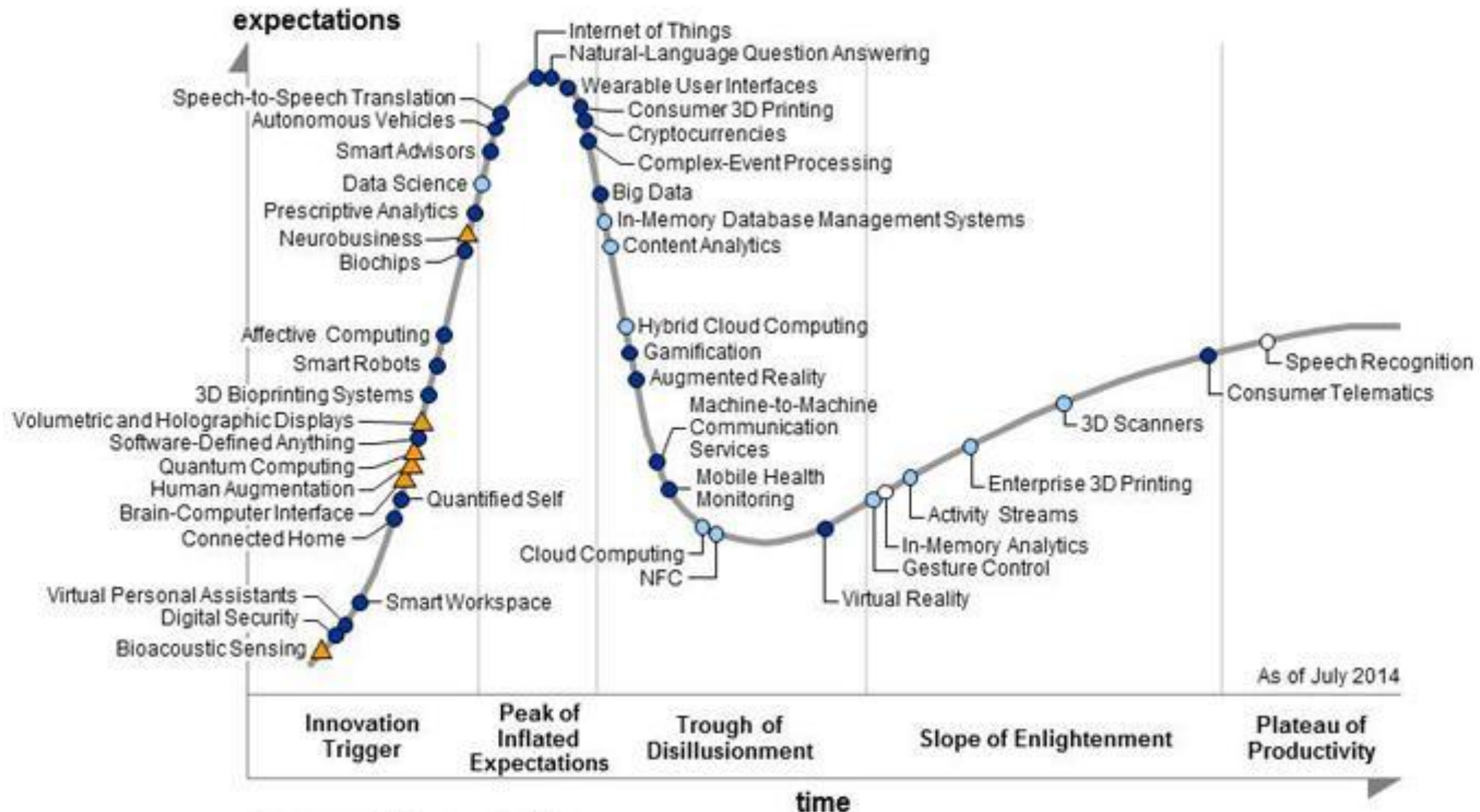
Plateau will be reached in:

- less than 2 years
- 2 to 5 years
- 5 to 10 years
- ▲ more than 10 years
- ⊗ obsolete before plateau



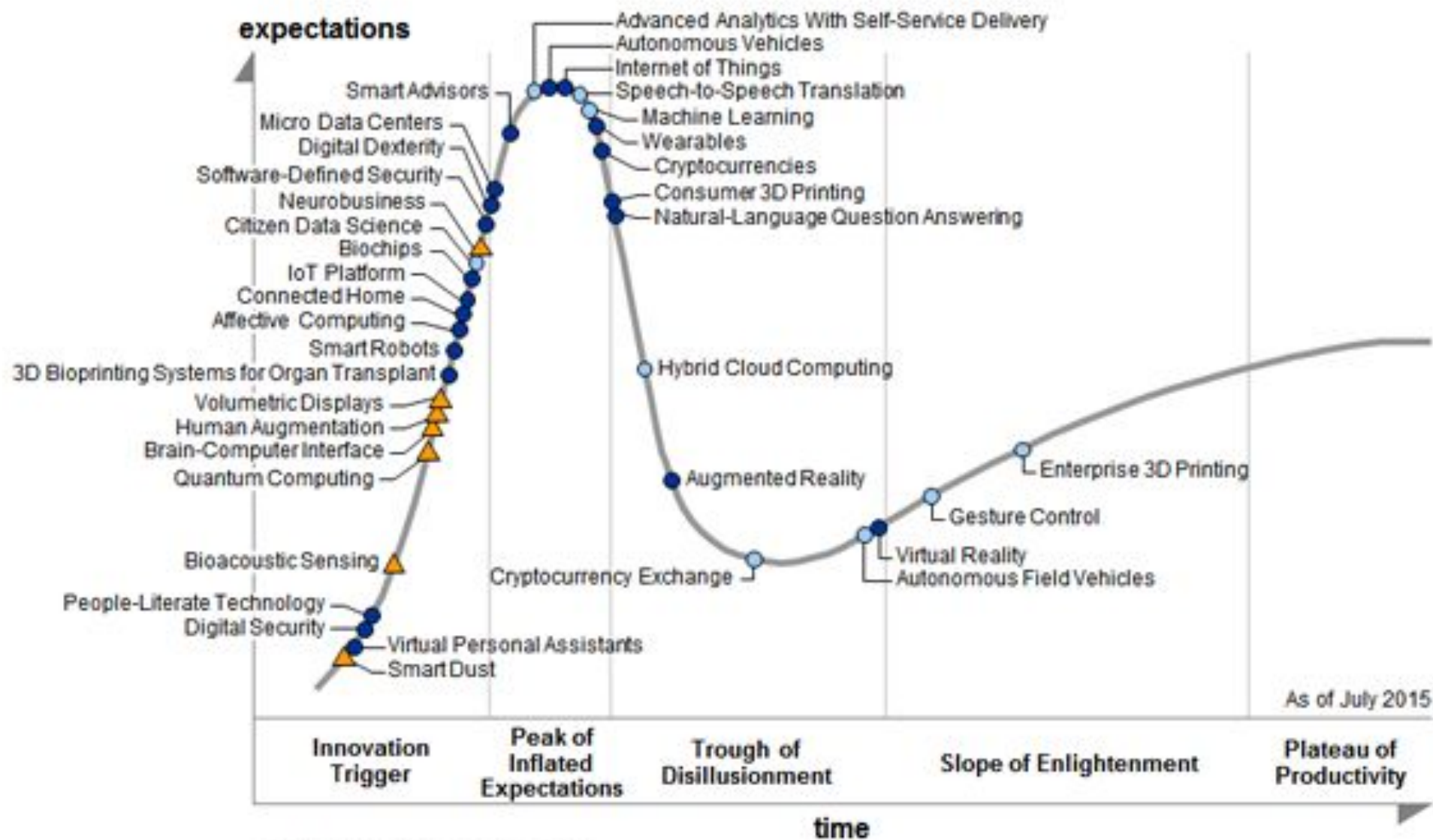
Plateau will be reached in:

- less than 2 years
- 2 to 5 years
- 5 to 10 years
- ▲ more than 10 years
- ⊗ obsolete before plateau



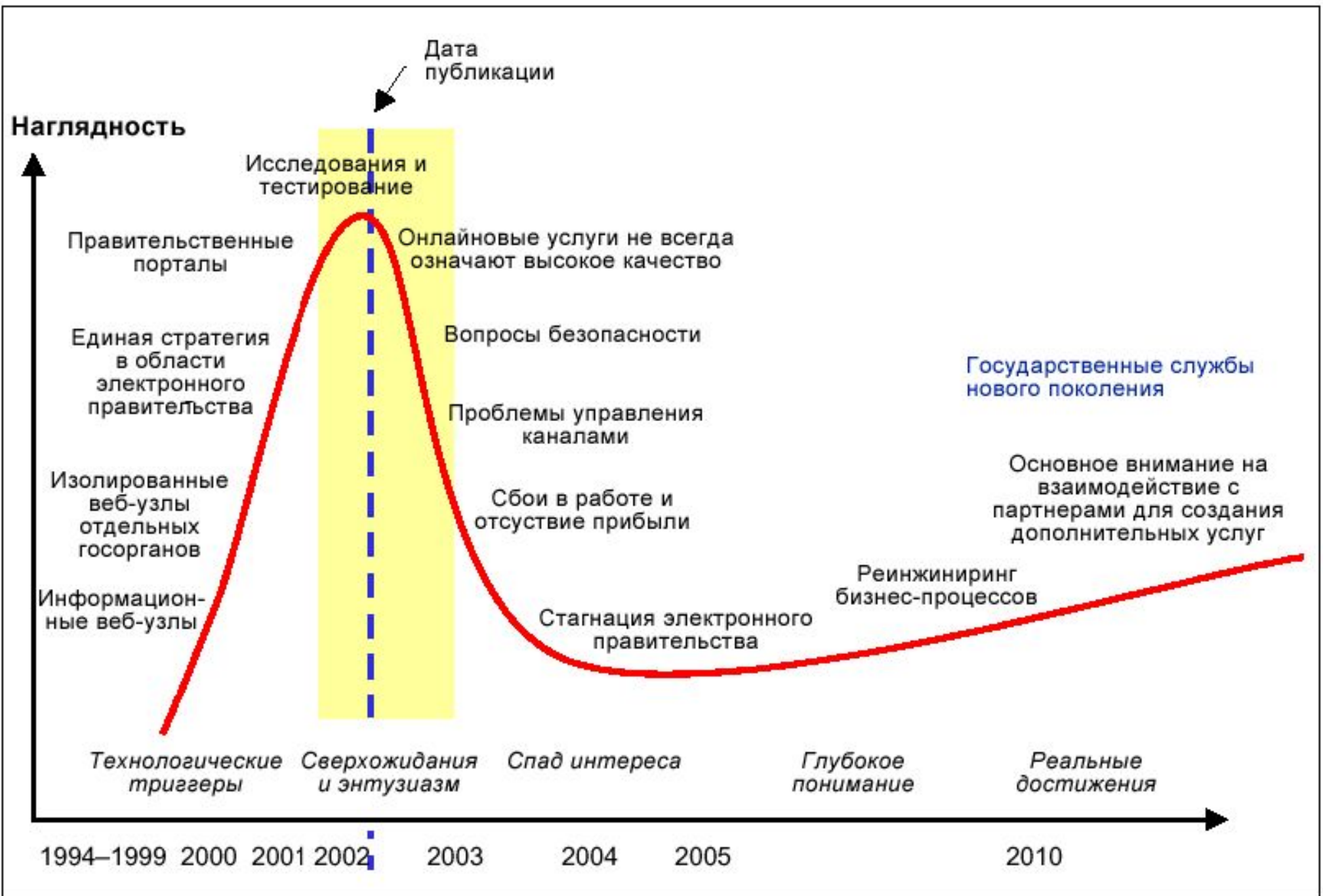
Internet of Things
 Natural-Language Question Answering
 Wearable User Interfaces
 Consumer 3D Printing
 Cryptocurrencies
 Complex-Event Processing
 Big Data
 In-Memory Database Management Systems
 Content Analytics
 Hybrid Cloud Computing
 Gamification
 Augmented Reality
 Machine-to-Machine Communication Services
 Mobile Health Monitoring
 Virtual Reality
 Activity Streams
 In-Memory Analytics
 Gesture Control
 Enterprise 3D Printing
 3D Scanners
 Consumer Telematics
 Speech Recognition

Speech-to-Speech Translation
 Autonomous Vehicles
 Smart Advisors
 Data Science
 Prescriptive Analytics
 Neurobusiness
 Biochips
 Affective Computing
 Smart Robots
 3D Bioprinting Systems
 Volumetric and Holographic Displays
 Software-Defined Anything
 Quantum Computing
 Human Augmentation
 Brain-Computer Interface
 Connected Home
 Quantified Self
 Cloud Computing
 NFC
 Smart Workspace
 Virtual Personal Assistants
 Digital Security
 Bioacoustic Sensing



Plateau will be reached in:

- less than 2 years
- ◐ 2 to 5 years
- 5 to 10 years
- ▲ more than 10 years
- ⊗ obsolete before plateau



Источник: Gartner Research

Web-приложения

Web-сеть (протоколы HTTP)

Сеть Интернет (протоколы TCP/IP)

Инфраструктура Интернет

– скоростные каналы передачи данных, мультиплексоры, маршрутизаторы, коммутаторы, сервера, специальное программное обеспечение, сетевые

Интернет - это сеть, состоящая из узлов, объединённых каналами связи



Серверы – пассивные программы, которые ожидают запросы от пользователей, обрабатывают их, формируют ответ и отправляют его пользователю

Клиенты – активные программы, с которыми обычно работают пользователи сети на своих компьютерах. Они отправляют запросы серверам для выполнения некоторой работы

Интернет-сокеты

– программные интерфейсы протокола TCP/IP, состоящие из комбинации IP адреса и номера порта от 0 до 65535.

http://8.8.8.8:8080/

Порты позволяют независимо использовать TCP-протокол на одном и том же компьютере несколькими программами одновременно.

Сервер при запуске сообщает операционной системе номер порта, который он хотел бы занять.

Система доменных имён

– система связывающая IP-адреса с символьными доменными именами, служащими для обозначения ресурсов Интернет в удобной для человека форме.

.ru – доменное имя первого уровня

usue.ru – доменное имя второго уровня

bi.usue.ru – доменное имя третьего уровня

Один IP-адрес может иметь много доменных имён.

Одному доменному имени может быть сопоставлено множество IP-адресов.

Адресация ресурсов web-сети

Uniform Resource Locator – уникальный идентификатор ресурса (URL – адрес)

<схема>://хост[:
порт]/путь/.../ресурс[?параметры
запроса][#ссылка]

Основная литература

3972.26

Т 81

Тузовский Анатолий
Фёдорович

Проектирование и
разработка web-
приложений: учеб. пособие
для академического
бакалавриата.

– М.: Издательство Юрайт,
2016. – 218 с.



<http://www.biblio-online.ru/book/198FC98A-BE39-4A85-B831-B6DCB3BBEE03>

echo

Выводит одну или более строк

```
void echo ( string $arg1 [, string $... ] )
```

`arg1` - выводимая строка

echo имеет также краткую форму, представляющую собой знак равенства, следующий непосредственно за открывающим тэгом. Например.

Добро пожаловать `<?= USER_NAME ?>` на сайт.

define

Определяет именованную константу

```
bool define ( string $name , mixed $value  
[, bool $case_insensitive = false ] )
```

name - имя константы

value - значение константы, допустимы скалярные величины и null. Допустимые скалярные значения: integer, float, string или boolean. Возможно определить константы типа resource, однако это не рекомендуется и может вызвать неопределенные последствия.

case_insensitive - если параметр установлен как TRUE, то константа будет определена без учета регистра. По умолчанию регистр учитывается, т.е. CONSTANT и Constant могут содержать различные значения.

```
<?php
define("USER_NAME", "Екатерина");
define("USER_AGE", "25");
define("USER_PHONE", "456-87-23");
define("USER_EMAIL", "kate@mail.com");
echo "Имя пользователя: USER_NAME";
?>

<p>Пользователю <?= USER_AGE ?> лет.</p>
<p>E-mail: <a href="mailto:<?= USER_EMAIL ?>">
<?= USER_EMAIL ?></a></p>
<p>Телефон: <a href="tel:<?= USER_PHONE ?>">
<?= USER_PHONE ?></a></p>
```

require_once

Включает и выполняет указанный файл. Если файл не находит, то останавливает выполнение скрипта. Проверяет, включался ли уже данный файл, и, если да, не будет включать его еще раз.

```
<?php
```

```
require_once 'config.php';
```

```
?>
```

