MAYER · BROWN

Software Development in a Dynamic Environment

Paul Chandler

Robert Kriss

Agenda

- **BACKGROUND**
  Modern software is developed in a dynamic environment

- **HYPOTHETICAL SCENARIO**
  Building a data scraping tool and website

- **ADDITIONAL CONTRACTUAL PROTECTIONS**

MAYER · BROWN

1

Background : Modern Software Development Happens in a Dynamic Environment

Modern Software Is Different:

- Much more complex than in the past (e.g., ERP, ecommerce, mobile apps)

- More interaction with external systems (e.g., outside websites, servers and IoT sensors)

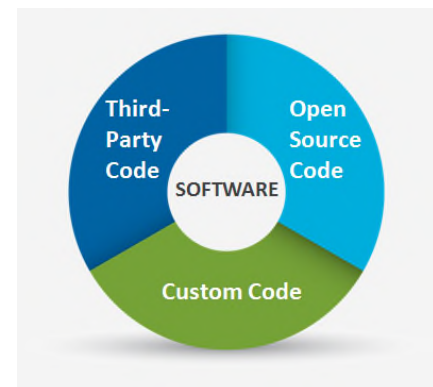- Technology is changing more rapidly
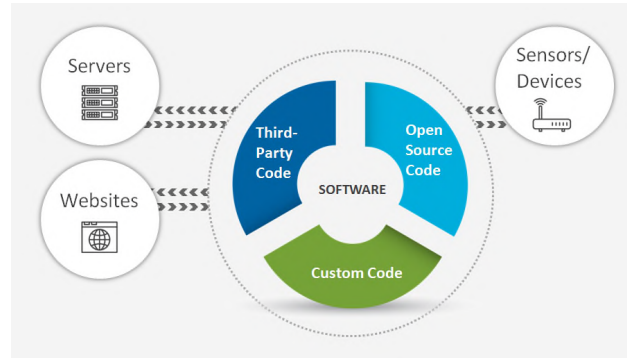
3

MAYER·BROWN

Background – Modern Software Development Happens in a Dynamic Environment

Modern Software Is Different

- Tighter project timelines and budgets

- Heavy reliance on third-party building blocks e.g., open source software, Amazon AWS/ Microsoft Azure SaaS platforms)

- Less custom-developed software

4

MAYER·BROWN

Background – What Makes the Development Environment Dynamic?

- **Building Block Changes:** Changes in open source software, SaaS offerings and other third-party building blocks used. For example:
    - Changes from new updates
    - Discovery of security vulnerabilities
    - Needed replacements due to loss of support
- **External Changes:** Changes in external systems that integrate with the developed software. For example:
    - Changes in client or supplier sites or data feeds

MAYER·BROWN

Background – What Makes the Development Environment Dynamic? – cont'd

- **Common themes of Building Block and External Changes –**
    - Not controlled by customer or system developer
    - May happen with no advance warning
    - May not be foreseeable at time of contract
    - Can have major adverse impacts on the project
- Often development contracts do not address the allocation responsibility for dealing with the impacts of these changes

MAYER·BROWN

Default Rule or Contract Provision?

- In the absence of an express allocation of risks in the contract, courts may apply a number of legal doctrines to decide disputes.

- In our discussion, we will see how some of those doctrines might be applied to a variety of hypothetical disputes.

- We will then see how many of these issues might be expressly addressed in the contract.

MAYER·BROWN

Legal Concepts

Legal Concept	Overview
Impracticability/Impossibility	Unanticipated events make continued performance under the contract impossible or unreasonably more expensive than expected.
Frustration of Purpose	Unanticipated events undermine the principal purpose for which one of the parties entered into the contract.
Mutual Mistake	Both parties entered into the contract under some shared mistaken (and material) belief of fact such that the contract really does not represent any true agreement between them.
Unilateral Mistake	One party entered into the contract under some mistaken (and material) belief of fact, and the other party knew or should have known of the mistake.

MAYER·BROWN

Impracticability/Impossibility (Basics)

Basics

- An unanticipated event made continued performance under the contract impracticable or impossible.

- The non-occurrence of the event was a "basic assumption" on which the contract was made.

- The party seeking to be excused from its obligations was not at fault for the impracticability or impossibility.

- The party seeking to be excused has not assumed the risk of the event.

New York Society for the Relief of the Ruptured and Crippled, Maintaining the Hospital for Special Surgery v. Wright Medical Tech., Inc., 2015 WL 4508358 (S.D.N.Y. Jul. 24, 2015)

MAYER·BROWN

9

Impracticability/Impossibility (Challenges)

Challenges

- What makes performance impracticable? How much money must be at stake?
  - Even if the costs would drive you bankrupt, that might not justify invoking the doctrine. *Sassower v. Blumenfeld*, 878 N.Y.S. 2d 602, 604 (N.Y. Sup. Ct. 2009)
  - But where the cost of addressing the changed circumstances eclipses the total contract price or the value of the assets at issue, courts have deemed them "excessive and unreasonable" and invoked the doctrine. *Asphalt Intern., Inc. v. Enterprise Shipping Corp., S.A.*, 667 F.2d 262, 265-66 (2d Cir. 1981).
  - No clear guidance at the end of the day.
- What is a "basic assumption" of the contract?
  - No clear guidance from the courts. Results are unpredictable.
- Did a party "assume the risk"?
  - Easy to tell when the contract expressly deals with the issue (e.g., indemnification clauses).
  - Few presumptive rules concerning implied assumption of risk.

MAYER·BROWN

10

Frustration of Purpose (Basics)

Basics

– An unanticipated event undermines or destroys (or "frustrates") a party's "principal purpose" in making the contract such that completing the transaction no longer makes any sense.

– The frustration "must be so severe that it is not fairly to be regarded as within the risks that he assumed under the contract".

– The non-occurrence of the frustrating event was a basic assumption on which the contract was made.

Gander Mountain Co. v. Islip U-Slip LLC, 923 F. Supp. 2d 351, 359 (N.D.N.Y. 2013)

11

MAYER·BROWN

Frustration of Purpose (Challenges)

Challenges

– What is a "basic assumption" of the contract?
  • No clear guidance from the courts. Results are unpredictable.
– Did a party "assume the risk"?
  • Easy to tell when the contract expressly deals with the issue (e.g., indemnification clauses).
  • Few presumptive rules concerning implied assumption of risk.
– What was the party's "principal purpose" in entering into the contract?
  • Often difficult to tell.
– Was the frustration "severe enough"?
  • No clear standards for assessing.

12

MAYER·BROWN
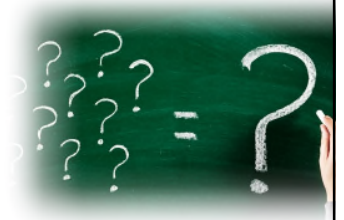
Mutual Mistake (Basics)

Basics

- Both parties shared the same erroneous belief as to a material fact, and so their acts in entering into the contract did not, in fact, accomplish their mutual intent.

- The mistake must be mutual, substantial and material and must exist at the time the contract was entered into.

ACA Galleries, Inc. v. Kinney, 928 F. Supp. 2d 699, 701 (S.D.N.Y. 2013)

13

MAYER·BROWN

Mutual Mistake (Challenges)

Challenges

- What is a "fact"?
    - Not always as clear as we might think (as we'll see)

- Was the mistake of fact "material" to the transaction?
    - Often difficult to tell except in the simplest of cases

14

MAYER·BROWN

Unilateral Mistake (Basics)

Elements

– One party entered into a contract under a mistake of material fact.

– The other party knew or should have known of the mistake.

– In some states, including New York, the non-mistaken party must have committed some underlying or associated fraud for certain sorts of relief to be available (e.g., reformation or rescission). However, if the mistake was as to a "basic assumption" of the contract, the contract may be voidable, even without any fraud.

Creative Waste Management, Inc. v. Capitol Environmental Services, Inc., 429 F. Supp. 2d 582, 599 (S.D.N.Y. 2006)

15

MAYER·BROWN

Unilateral Mistake (Challenges)

Challenges

– Again, what is a "fact"?

– Was the mistake "material"?

– Was there underlying fraud? Can you prove it?

– Did the mistake concern a "basic assumption"?

16

MAYER·BROWN

Hypothetical Scenario

- A company wants to deploy a new website to display aggregated airline pricing information from other sources to its customers.

- A software vendor is retained to develop a custom "bolt on" enhancement to "Aggregator Version 1," an open source data aggregation software package. The enhancement will scrape data from specified airline websites, pass the data to the Aggregator and then onto the website, which the developer will also build.

- The software contract specifies the use of Aggregator Version 1, specifies the parameters of the websites to be scraped, contains a 90-day warranty the software will scrape the websites in question and an additional warranty that the services will be performed in a good and workmanlike manner consistent with industry standards.

17

MAYER·BROWN

Hypothetical Scenario

18

MAYER·BROWN

9
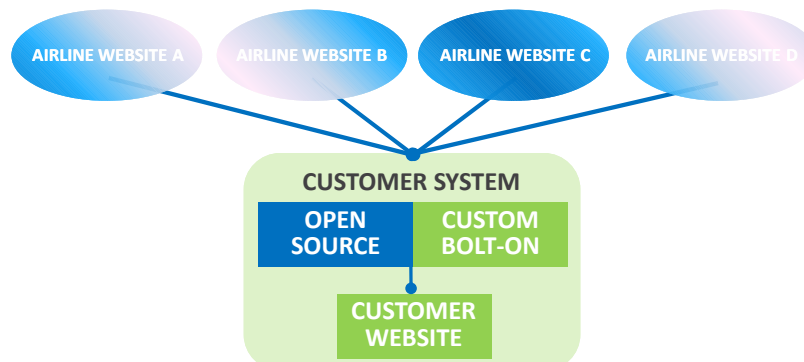
Issue 1: Who Is Responsible for Addressing Security Vulnerabilities in Open Source Code?

- During development, a security vulnerability in Aggregator Version 1 is publicly disclosed. Aggregator Version 2 has just been released and does not contain the vulnerability.

- However, developing the "bolt on" enhancement and the website will be considerably more difficult and expensive if Version 2 is used.

MAYER·BROWN

Contracting Lessons

- **Specify the required results** rather than the technical means of achieving them
  - But: In some cases, consider specifying particular coding standards to be followed

- **Address responsibility and cost** for identifying, monitoring and remedying security vulnerabilities in the initial contract

- **Consider specifying an objective definition** of what constitutes a "security vulnerability," and require that such vulnerabilities not be in the final product

- **Consider penetration testing** by a designated consultant as a part of acceptance testing

- **Perform due diligence** on third-party building blocks before the project begins

MAYER·BROWN

Issue 2: Who is Responsible for Changes in External Systems before Delivery of Software?

- The developer's work proceeds for a time. However, before the project is completed, the developer finds that several of the websites to be scraped have been reengineered.

- The custom "bolt on" enhancement as developed to date is now incompatible with these sites and cannot scrape the required pricing data.

21

MAYER·BROWN

Contracting Lessons

- Tie acceptance to delivery of required business results rather than compliance with technical parameters

- Require supplier to conduct due diligence on third-party building blocks and external systems before coding begins

22

MAYER·BROWN

Issue 3: Who Is Responsible for External Systems Following Delivery of Software?

- 120 days after Go Live, two more websites are reengineered, and the scraping module is no longer compatible

- The company consults an expert who informs them that the software could have been designed and coded to make it much less expensive and time-consuming to address the changes in the two sites

23

Contracting Lessons

- **Consider** including specific code quality requirements to avoid relying solely on "industry standards" and required business results

- **Conduct** due diligence on supplier and its proposed developers to determine their level of skill and quality (in addition to qualified personnel warranties)

- **Negotiate** longer warranty periods:

  – Expands opportunities to find latent defects

  – Duration may be driven by external system business cycles that could trigger additional changes

24

Additional Contractual Protections

- Customers may mitigate risk by doing the following:

  – Push for liberal termination rights, along with:

    • A right to receive the software in both object and source code forms

    • A broad license to the developer's IP to continue to use and modify the software

  – Secure the right to hire (or at least to receive knowledge transfer from) the developer's personnel working on the project

  – Avoid giving the developer exclusive rights to develop the software or other commitments that may restrict the business from engaging a successor developer

25

MAYER · BROWN

QUESTIONS?

Paul Chandler
Counsel

+1 312 701 8499
pchandler@mayerbrown.com

Robert Kriss
Partner

+1 312 701 7165
rkriss@mayerbrown.com

MAYER · BROWN